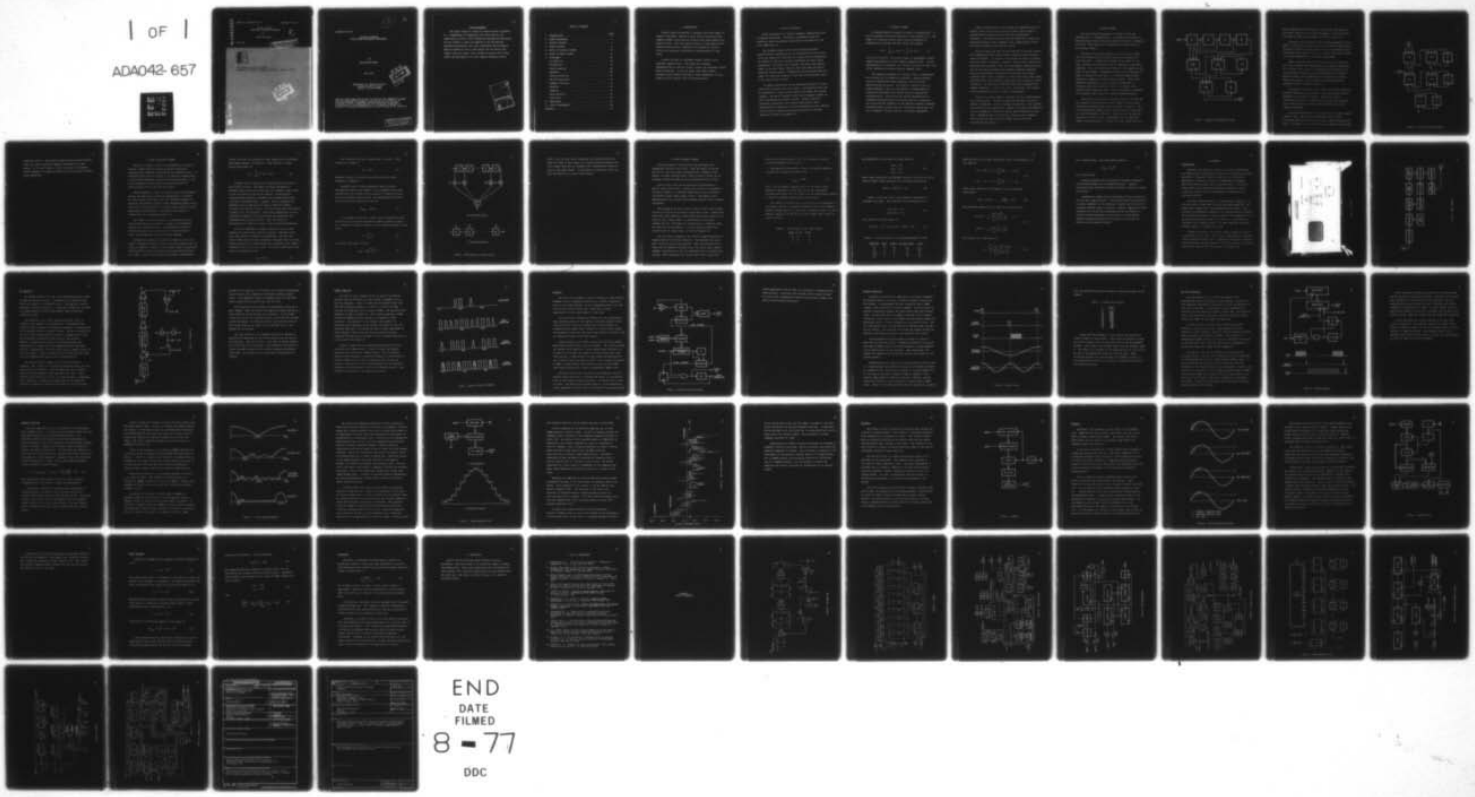
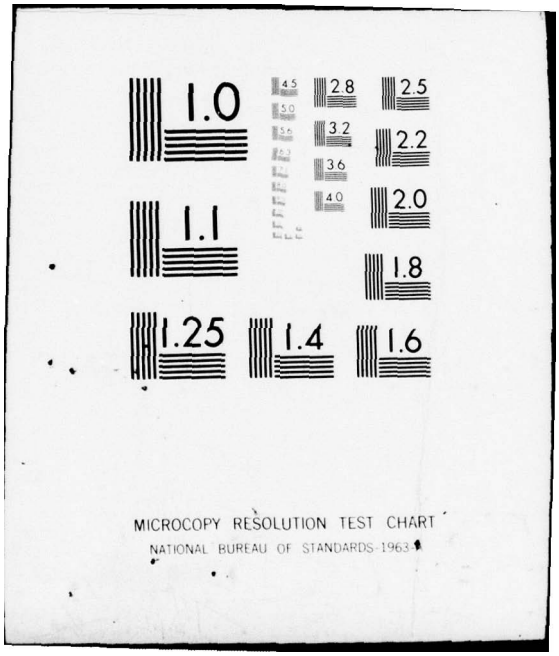
AD-A042 657 ILLINOIS UNIV AT URBANA-CHAMPAIGN DEPT OF COMPUTER SCIENCE F/G 9/4
DIGITAL FILTERING USING BURST PROCESSING TECHNIQUES.(U)
JUL 77 D K WELLS N00014-75-C-0982
UNCLASSIFIED UIUCDCS-R-77-871 NL

1 OF 1
ADA042-657



END
DATE FILMED
8-77
DDC


1.0
4.5
5.0
5.6
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.8
1.1
1.25
1.4
1.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

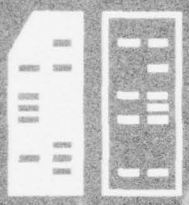
Report No. UIUCDCS-R-77-871 UILU-ENG 77 1736

DIGITAL FILTERING USING BURST PROCESSING TECHNIQUES

by

DAVID KEVIN WELLS

July 1977



DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN URBANA, ILLINOIS

(5)

ii

UIUCDCS-R-77-871

DIGITAL FILTERING USING BURST PROCESSING TECHNIQUES

BY

DAVID KEVIN WELLS

July 1977

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

This work was supported in part by Contract No. N00014-75-C-0982 and the Department of Computer Science at the Universtiy of Illinois, Urbana- Champaign, and was submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering, at the University of Illinois.

(See form 1473)

ACKNOWLEDGEMENT

The author wishes to thank his thesis advisor Professor W. J. Poppelbaum for suggesting this topic and for the opportunity to work in the Information Engineering Laboratory. He also wishes to thank the members of the Information Engineering Laboratory for their assistance and friendship. Special thanks are due to Gary Taylor for his advice and support when the author first joined the group, and to Gary Gostin and MartinWolff for their expert technical advice.

TABLE OF CONTENTS

1. INTRODUCTION

Digital signal processing is becoming ever more common in today's technology. Digital filters can be constructed with complex transfer functions and selectivities unattainable with analog filters. With the rapid increase of integrated circuit expertise making their components less expensive, digital filters are replacing traditionally analog designs in many areas.

Finding new ways to implement digital filters is an important research area. This paper will discuss implementations of digital filters using a new technique called Burst Processing. It will be shown that these filters, designed using standard techniques, offer advantages in their simplicity and complete lack of synchronization.

2. Burst Processing

Burst Processing is a unitary weighted, asynchronous form of digital processing. Retaining a reasonable degree of precision, Burst Processing enjoys much of stochastics's low cost simplicity.[1]

The encoding scheme used in Burst Processing makes synchronization *unnecessary in the* receiving of a burst value. Given any consecutive N periods of a serial burst transmission, the number of "one bits" present represents the burst value transmitted. The basic burst decoder, the block sum register (BSR), is an N-bit shift register with each stage driving an identical current source. Summing the currents determines the number of "one bits" present in the last N time periods, which is the burst value received.

Arithmetic operations can be performed digitally in burst using shift registers and logic gates. Alternatively they may be performed in a quasi-analog fashion with BSR's and current summing. The use of unequal current sources opens a number of other interesting processing avenues. Several projects, including a calculator and a digital AM receiver, have demonstrated the feasibility of Burst Processing.[2-4] Several other projects ranging from spectrum analysis to failsoft memories are under development.[5]

3. DIGITAL FILTERS

An understanding of digital filtering is assumed here. It can be obtained from one of the texts on the subject.[6,7] The output of a general digital filter can be expressed as a weighted sum of present and past inputs and outputs:

$$y(n) = \sum_{i=1}^{M} a_i \, x(n-i) + \sum_{i=1}^{N} b_i \, y(n-i) \qquad (1)$$

If all b_i's are zero, the filter output is independent of past outputs and exhibits a finite impulse response (FIR). If the implementation is recursive i.e. the b_i's are not all zero, the filter exhibits an infinite impulse response (IIR).

The frequency response of a digital filter is determined by the choice of its weighting constants. Several methods exist for determining these constants, when the filter specifications are given. The design of analog filters is highly advanced. Thus for IIR designs a transformation is customarily performed on an analog design of appropriate characteristics. FIR filters are usually designed with a windowing technique where a smoothing function is applied to the finite Fourier Transform of the desired frequency response. Achieving better results than the traditional approaches, the art of computer iterative design is becoming commonplace.

A major consideration in the design and implementation of a digital filter is the finite precision which must be used in representing the weighting constants and in performing arithmetic. Deviations from the desired coefficient values can drastically effect filter response, while computational errors can cause instability and introduce noise.

There are pros and cons for both FIR and IIR digital filters when implemented using either fixed or floating point weighted binary arithmetic. In general, IIR filters are easier to design and less costly in hardware. However, they are very sensitive to quantized pole placement, have non-linear phase, characteristics, and are not easily designed with complex frequency responses. They are also subject to cumulative errors and limit cycles caused by their finite arithmetic. FIR filters have linear phase characterisics, are inherently stable and, are tolerant of finite arithmetic. By iteration techniques they can be designed with complex responses. On the other hand, they suffer from complex design procedures and large hardware requirements.

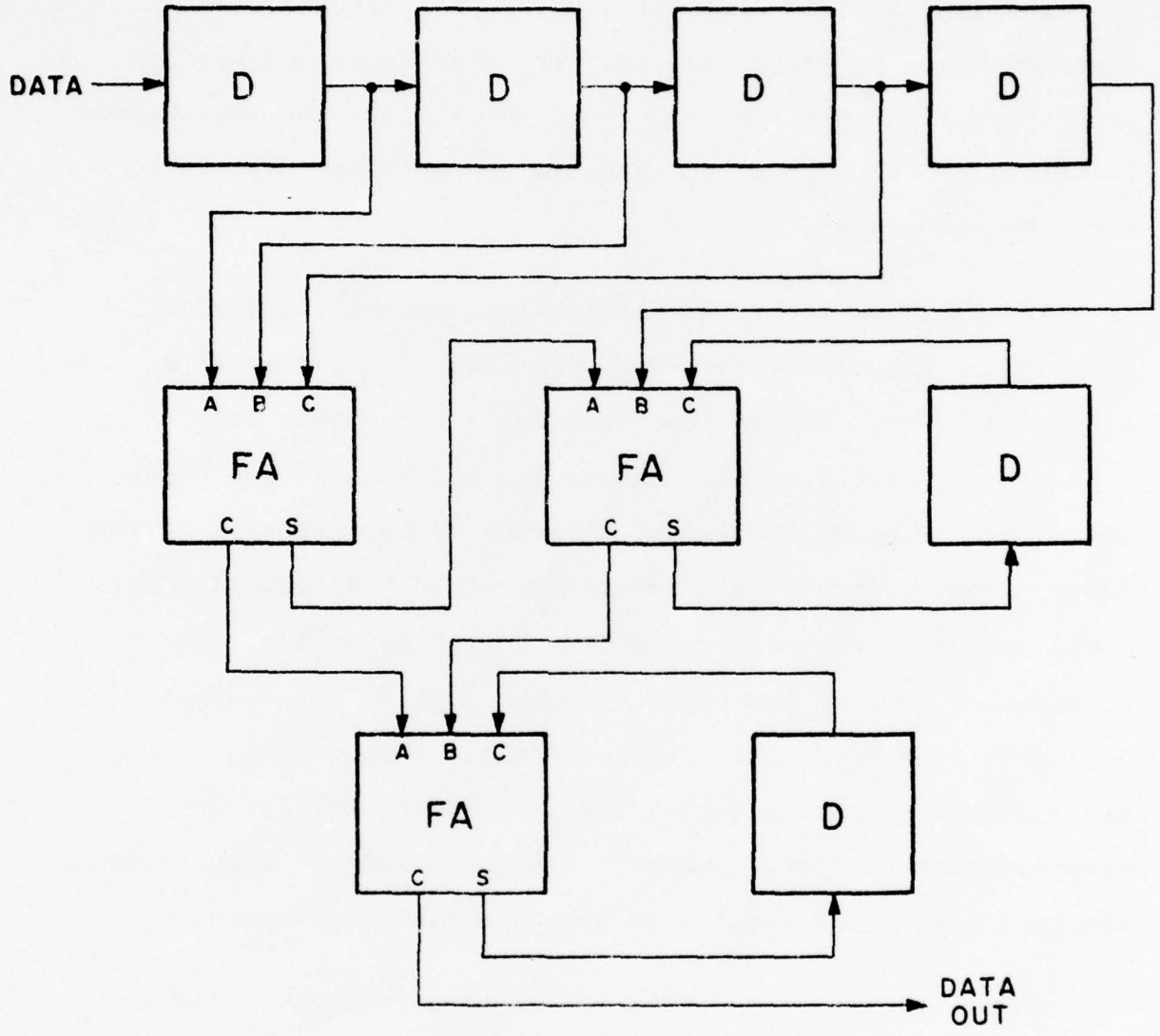
There are a large number of possible circuit configurations for a given filter. The most frequently used fall into three general catagories: direct, cascade, and parallel. The transfer function of all are equivalent. However, the noise introduced by finite arithmetic in the filter may be quite different in each case. Depending upon the precision, the particular response desired, and the type of filter chosen any of the three realizations may be best.

4. BURST FILTERS

The use of Burst Processing in digital filters was proposed by W. J. Poppelbaum.[8] Its simplicity and lack of synchronization create attractive possibilities for inexpensive filtering. Gary Taylor [9] extended the original idea to a more general design.

The basic burst filter element is a one-bit full adder (FA) and an associated flip-flop referred to together as a perverted adder. Connecting the adder's sum output to its input through the flip-flop allows the addition of two burst sequences. The sum, scaled by a factor of two, appears at the adder's carry output. Remainders stored in the flip-flop are added in during the next cycle. In filtering applications consecutive samples are often closely related. Thus, while appearing unfortunate at first, using the remainder in succeeding additions actually results in a better filter response than truncation would. Also, discarding or rounding the remainder would require an unwanted synchronization.

Burst filters need not be scaled to prevent overflow and will maintain the precision of their inputs. A four stage rectangular weighted burst filter is shown in Figure 1. The tree configuration common to all burst filters is referred to as a perverted adder or PA tree. Each level in the PA tree scales by a factor of two, a consequence of the perverted adder's scaling property. As there are four inputs from the



Filter 1. Rectangular Weighted Burst Filter

shift registers and two levels in the PA tree, the output of the filter remains an N bit burst encoding. Precision can be increased by reducing the number of levels in the tree.

Arrising from the factor of two scaling performed by every adder in a PA tree, each level in the tree can be thought of as carrying an effective weight double that of the previous level. Allowing PA tree inputs to enter at various levels thus results in an easy implementation of integer weightings!

Number representation in burst filters can be offset by N/2, thus allowing easy implementation of negative quantities. For a ten-bit burst, zero is represented by five ones out of ten time slots, negative five is represented by all zeroes. Negative weights are implemented by the use of an inverter before tree inputs. A three level tree showing multiple weighting values is shown in Figure 2.

Some design freedom is lost because only a small number of possible weightings are available: this limits pole and zero placement in the filter response. Large trees can be constructed giving more weightings, but quite good responses can be obtained with a limited number of weightings using computer techniques.

For Burst Processing the filter must be of a fixed impulse response type. Low precision arithmetic used in Burst Processing prohibits the use of a filter response dominated by poles. Arithmetic errors would result in instability and large

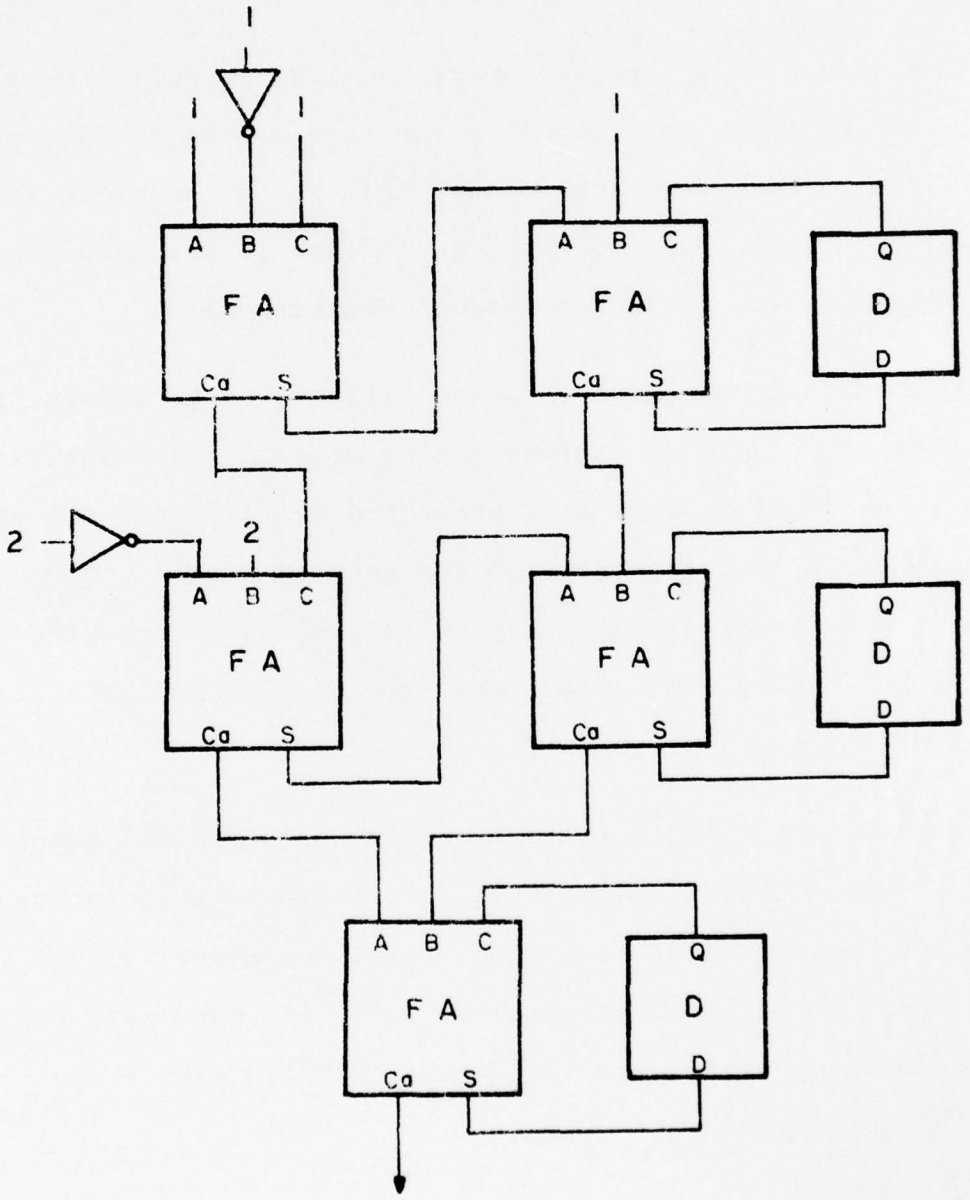


Figure 2. PA Tree With Multiple Weights

cumulative errors. And limited weighting values would severely limit the useful frequency responses attainable in a pole design. As in other digital filters the type of FIR design chosen depends on a number of factors with no particular choice being predominant.

5. NOISE IN DIGITAL FILTERS

Noise in a digital filter is introduced by a variety of sources. Among these are finite precision coefficient representation, sampling, quantization and roundoff errors. In analyzing these sources of noise it is generally assumed that: 1) each source generates uniformly distributed errors, 2) the noise introduced is white and 3) the noise sources are uncorrelated with the input and each other.

These assumptions, while not always valid, allow seperate analysis of each noise source and superposition of results. For most filters with inputs that are reasonably dynamic in amplitude and spectral content, these assumptions appear to be valid with filter precision of at least eight binary bits.[10] Experimental justification is attainable with inputs as deterministic as a sinusoidal signal.[11]

The effect of finite precision in representing filter coefficients should be viewed as a change in the frequency response of the desired filter. By appropriate choice of coefficients under the restrictions of finite precision a filter can be designed with near optimum response.

Assuming that sampling is above the Nyquist rate of the highest frequency present, no error will be introduced into the signal in sampling. However, as the signal cannot be perfectly represented by a finite precision device quantization errors will result. Given S as the step size between representable

values, the error in encoding an input signal will be uniformly distributed between -S/2 and S/2. This results in a mean square noise power of:

$$N_Q = \int_{-S/2}^{S/2} x^2 \, dx = s^3/12 \qquad (2)$$

Roundoff errors represent the largest source of noise in most digital filters. The amount of noise introduced is dependent upon the precision and type of arithmetic used as well as the realization of the filter network. Floating point representations generally introduce less roundoff noise than fixed point representations. However, they are expensive to implement and are generally not used in specific hardware realization of digital filters. For this reason only fixed point arithmetic will be discussed. FIR filters implemented in two's complement arithmetic more nearly match the properties and implementations of burst filters. They are chosen as representative of digital filters in the following discussions.

For two's complement arithmetic addition is error free assuming the final result does not overflow. Roundoff errors occur when two N-bit numbers are multiplied. The resultant 2N-bit number must be either rounded or truncated back to N bits. Assuming an even error distribution and a stepsize S the range of rounding error will be -S/2 to S/2. As this is identical to the range of quantization error,

$$N_R = S^3/12 \qquad (3)$$

For truncation the error ranges from -S to zero. This results in a noise of

$$N_T = S^3/3 \tag{4}$$

Rounding is seen to introduce less noise but may be more expensive to implement.

Roundoff noise is highly dependent upon the filter configuration used in the design. For a standard direct form realization of an M stage filter as shown in Figure 3a, there will be M roundoff noise sources. By superposition the total noise at the filter output will then be

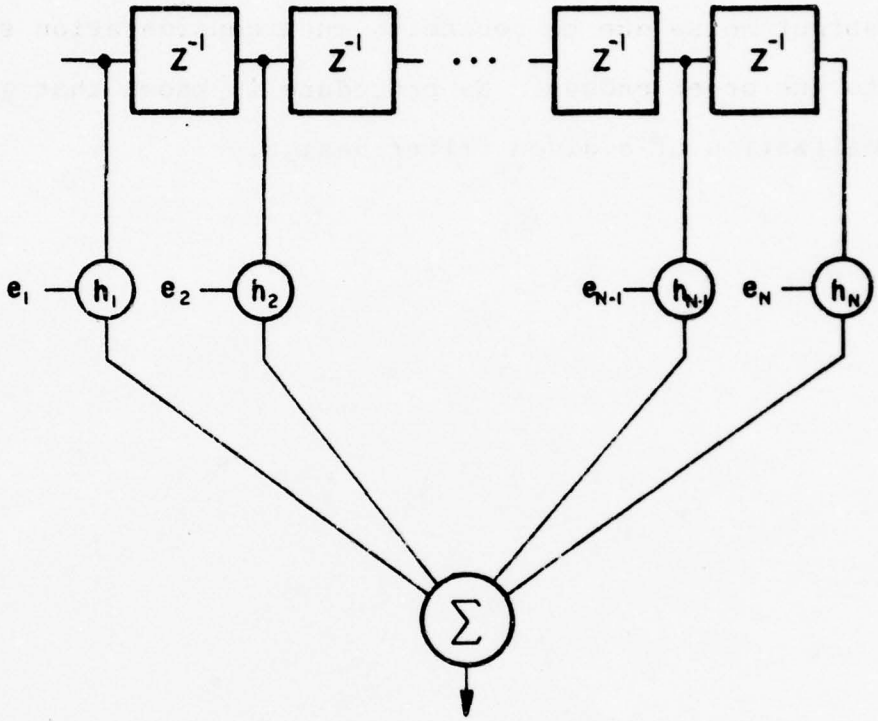
$$N_{Total} = MS^3/12 \tag{5}$$

If a cascade realization is used, noise introduced in each stage will be filtered by successive stages as shown in Figure 3b. Assuming an impulse response for the remaining stages in the filter

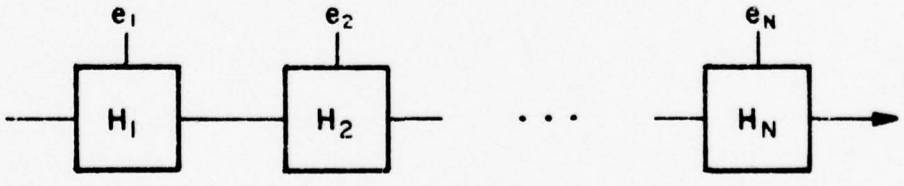
$$h = \sum_{i=0}^{M-1} h_i \tag{6}$$

the output noise power will be

$$N_{out} = kN_R \sum_{i=0}^{M-1} h_i^2 \tag{7}$$



(a) Direct Realization



(b) Cascade Realization

Figure 3. Noise Sources In Digital Filters

where k is a scaling factor necessary for preventing overflow. Since the order of the stages in a cascade design determines the total output noise due to roundoff, much consideration should be given to the order chosen. No procedure is known that gives the best realization of a given filter design.

6. NOISE IN BURST FILTERS

Burst Processing's low precision may invalidate the assumptions of white error noise. Also the effect of stored carries in a PA tree cannot be analyzed as a roundoff noise. Subject to these considerations, noise in burst filters can be analyzed in much the same way as for weighted binary filters.

Burst filter noise can be analytically determined for special case filters only. Stored carry noise is correlated by successive stages of a cascaded design and is even internally correlated in many single stage filters. Fortunately good approximations can be made using standard digital filter analysis techniques.

The assumption of white noise in burst filters can be shown to hold if certain non-stringent conditions are met. Bennett has shown that after sampling a bandlimited white noise signal at an infinite rate, signal power is concentrated in the lower frequencies.[12] The degree of concentration is dependant upon the step size of the sampler. A large step size causes more concentration of signal power in the low frequencies.

For non-infinite sampling, the infinite sampled spectrum is wrapped around the folding frequency. This spreads the signal power uniformly across the spectrum. When the quantization is rough and the sampling frequency high, the signal power remains somewhat concentrated in the low frequencies due to insufficient folding. Noise generated in a filter under these conditions

would also be concentrated in the low frequencies causing correlation between noise sources.

Extrapolating from Bennett's data, the maximum sampling frequency for a signal is found to be:

$$f_{s_{max}} = 1.5Bf \qquad (8)$$

where f is the signal frequency and B is the burst length. Equation 8 expresses in more exact terms the requirement discussed previously that the signal must be sufficiently dynamic in nature to justify assuming white noise errors.

The effect of stored carry in a PA tree can be analyzed for the special case of a rectangular weighted filter, For example: Assume a random input to a two stage rectangular filter, the four possible inputs to the PA tree can then assume three values as shown in Table 1.

Table 1. A Two Stage Filter Input Values

Input (x ,x)		Value
0	0	0
0	1	1
1	0	1
1	1	2

The probability of occurence for each value is:

$$\begin{aligned} p(0) &= 1/4 \\ p(1) &= 1/2 \\ p(2) &= 1/4 \end{aligned} \tag{9}$$

Under these conditions the remainder stored in the PA tree can be shown by Markov chain analysis to be uniformly distributed.

$$p(R=0) = p(R=1/2) = 1/2 \tag{10}$$

Table 2 lists the error for all possible conditions of remainder and input. The resultant error distribution is:

$$\begin{aligned} p(|e|=0) &= 1/2 \\ p(|e|=1/2) &= 1/2 \end{aligned} \tag{11}$$

This results in a noise power of:

$$p(|e|=0) \cdot 0^2 + p(|e|=1/2) \cdot (1/2)^2 = 1/8 \tag{12}$$

Table 2. Error Values For A Two Stage Rectangular Filter

Remainder	Input	Output	Desired Output	Error
0	0	0	0	0
0	1	0	1/2	-1/2
0	2	1	1	0
1/2	0	0	0	0
1/2	1	1	1/2	1/2
1/2	2	1	1	0

Generalizing to an M stage rectangular filter the probabiltiy of any input is:

$$p(I) = \binom{M}{I} \Big/ \left(M + \sum_{K=0}^{M-1} \binom{M}{K} \right) \qquad I \neq 0,M$$

$$p(I) = M \Big/ \left(M + \sum_{K=0}^{M-1} \binom{M}{K} \right) \qquad I = 0,M \tag{13}$$

Under these conditions the remainder will be uniformly distributed.

$$p(0) = p(1/M) = \ldots = p\left(\frac{M-1}{M}\right) = 1/M \tag{14}$$

The resultant probability for each error value will be:

$$p(e=I/M) = \frac{(M-I)\binom{M}{I}}{M + \sum_{K=0}^{M-1} (M-K)\binom{M}{K}} \qquad I \neq 0$$

$$p(e=0) = \frac{M}{M + \sum_{K=0}^{M-1} (M-K)\binom{M}{K}} \tag{15}$$

This results in a noise power of

$$N_s = \frac{\sum_{I=1}^{M-1} (M-I)\binom{M}{I}(I/M)^2}{M + \sum_{K=0}^{M-1} (M-K)\binom{M}{K}} \tag{16}$$

for a one-bit burst. The noise power would be

$$N_{out} = N_s/B^2 \qquad (17)$$

for a B-bit burst.

Similar equations can be derived for M-stage rectangular filters which output with increased precision. Computer simulations have verified the above analytical results to within one percent.

Cascading of stages and non-rectangular filters proved too dificult for simple analysis. Correlation between noise sources is highly dependent on the ordering of stages and the choice of weighting patterns used in a design. However, adapting the above equations for non-uniform weightings and the use Equation 6 allows for good estimation of the noise in a given design. Computer simulation shows this estimation to be within ten percent for all designs studied.

7. LOGITUNER

Introduction

Logituner was chosen as a project in order to demonstrate the feasibility of using Burst Processing in filtering and signal processing. Logituner, shown in Figure 4, is an AM radio employing all digital circuitry in transforming a wideband amplified antenna signal into the desired audio. Utilizing a fixed system clock and Burst Processing techniques, tuning, filtering, volume control, and bandwidth selection are all achieved digitally.

The basic system structure is illustrated in Figure 5. An antenna signal is amplified by an AM bandlimited amplifier and delta block encoded into a non-compacted burst signal. This encoding is subsequently sampled at the desired station frequency by the station selector. A direct result of this sampling is the shifting of the station down to DC, where it is extracted by a cascade of four lowpass filters. An audio amplifier sends the decoded signal to a speaker for output.

Further filtering of the station signal yields its carrier in the form of a DC level. This level is monitored by a feedback loop to maintain station lock. Completing the design a beatdown block interfaces the high speed ECL circuitry used in the encoder and station selector with the system's low speed TTL.

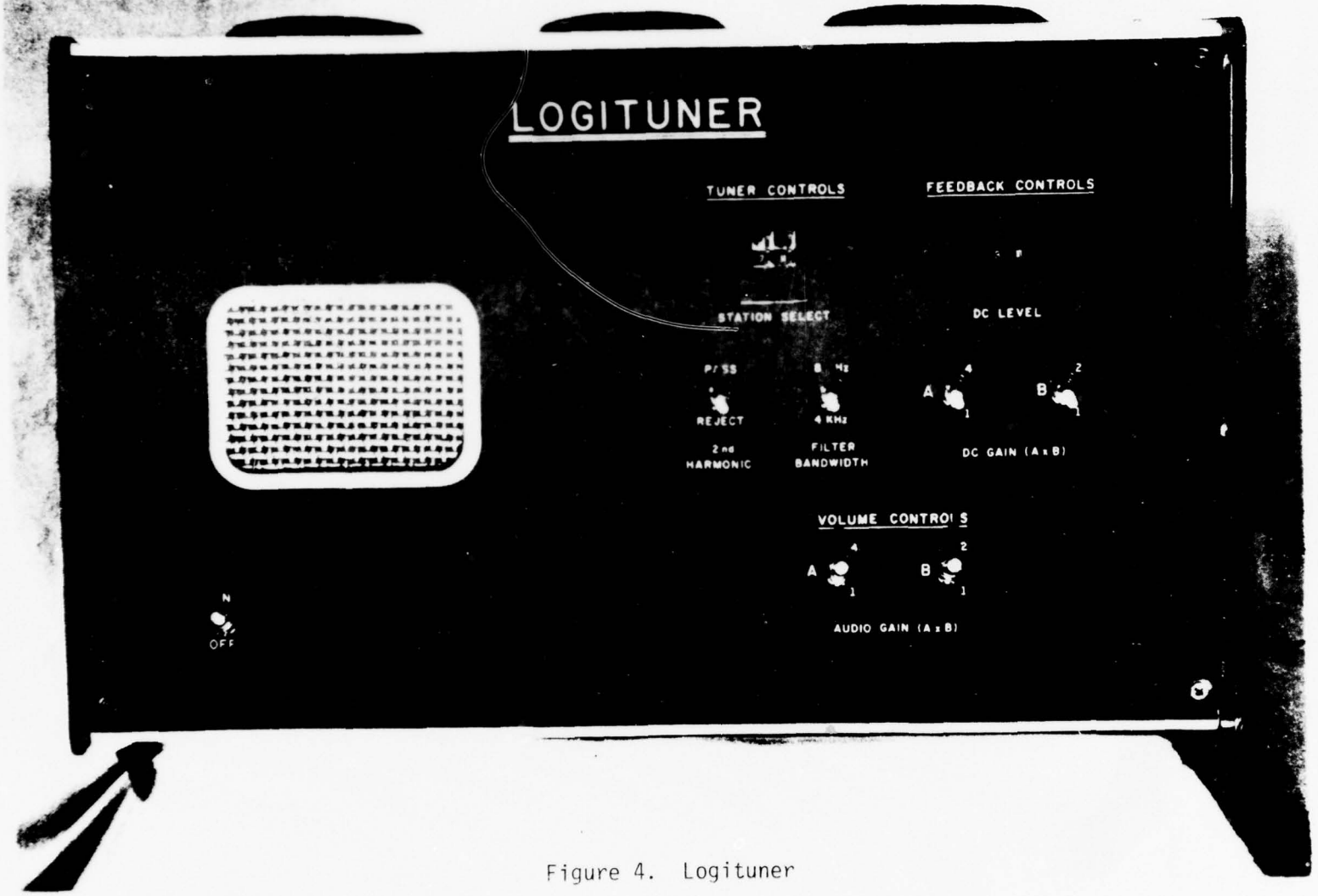


Figure 4. Logituner

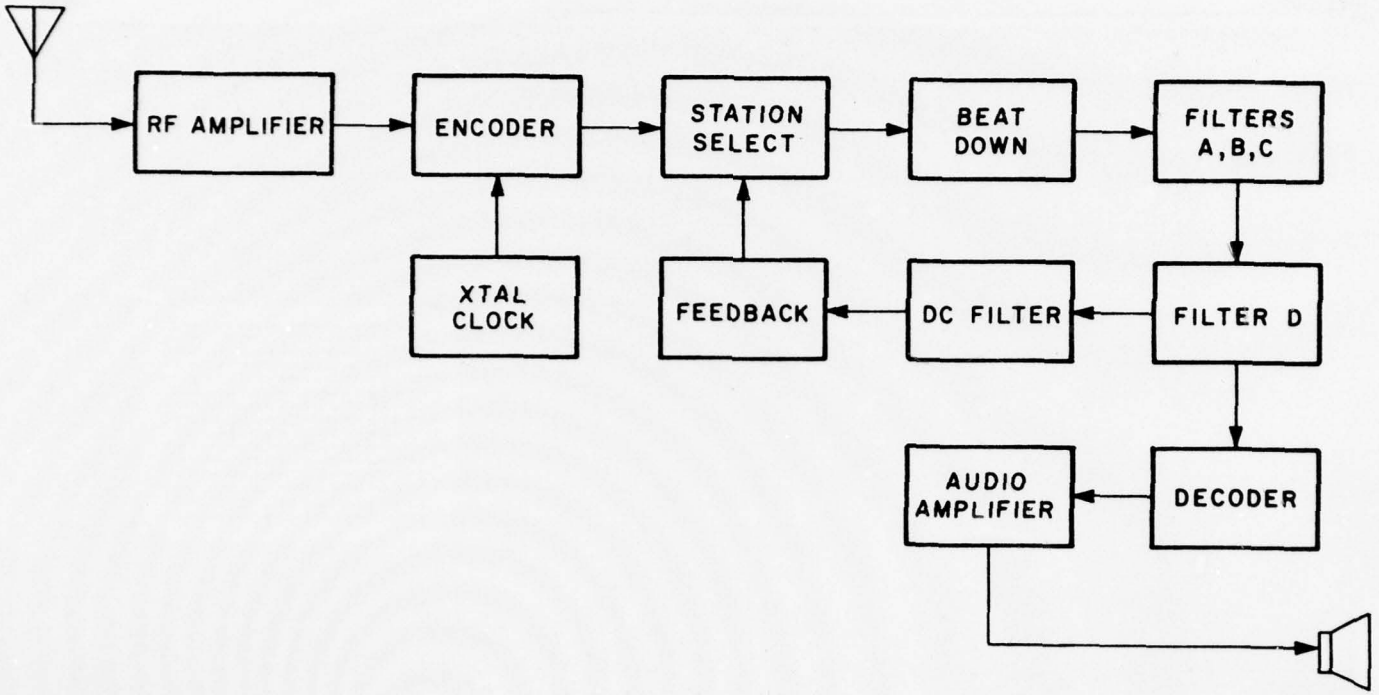


Figure 5. Block Diagram

RF Amplifier

An antenna signal of at most a few hundred microvolts cannot be used by digital circuitry. Consequently, an amplifier must boost this signal to a reasonable level. The amplifier used in Logituner is shown in Figure 6. Prevention of amplifier overload is accomplished by the multistage design, which guaranties stability of gain.

The amplifier uses a three stage sixth order filter to bandlimit its response to AM frequencies. The bandlimiting, chosen to allow reception of commercial stations, is necessitated by the periodic nature of digital filters. An obvious choice for the first filter stage should have been a highpass filter which would eliminate everpresent sixty cycle AC interference. However, in testing this design, the beginning stages were overpowered by a strong local two megahertz single side band station. Consequently, the final design uses a lowpass filter as the first stage. Overall response of this filter was quite good with flat passband and steep rolloff characteristics.

The system encoder accepts a two volt input without clipping. Best results are obtained by using the entire two volt range as much as possible. Thus the amplifier has a feedback loop which keeps maximum output peaks near two volts. This feedback loop consists of a pair of current sources, a comparator and a capacitor. A buffered voltage stored on the capacitor controls the amplifier through an AGC input on its first stage.

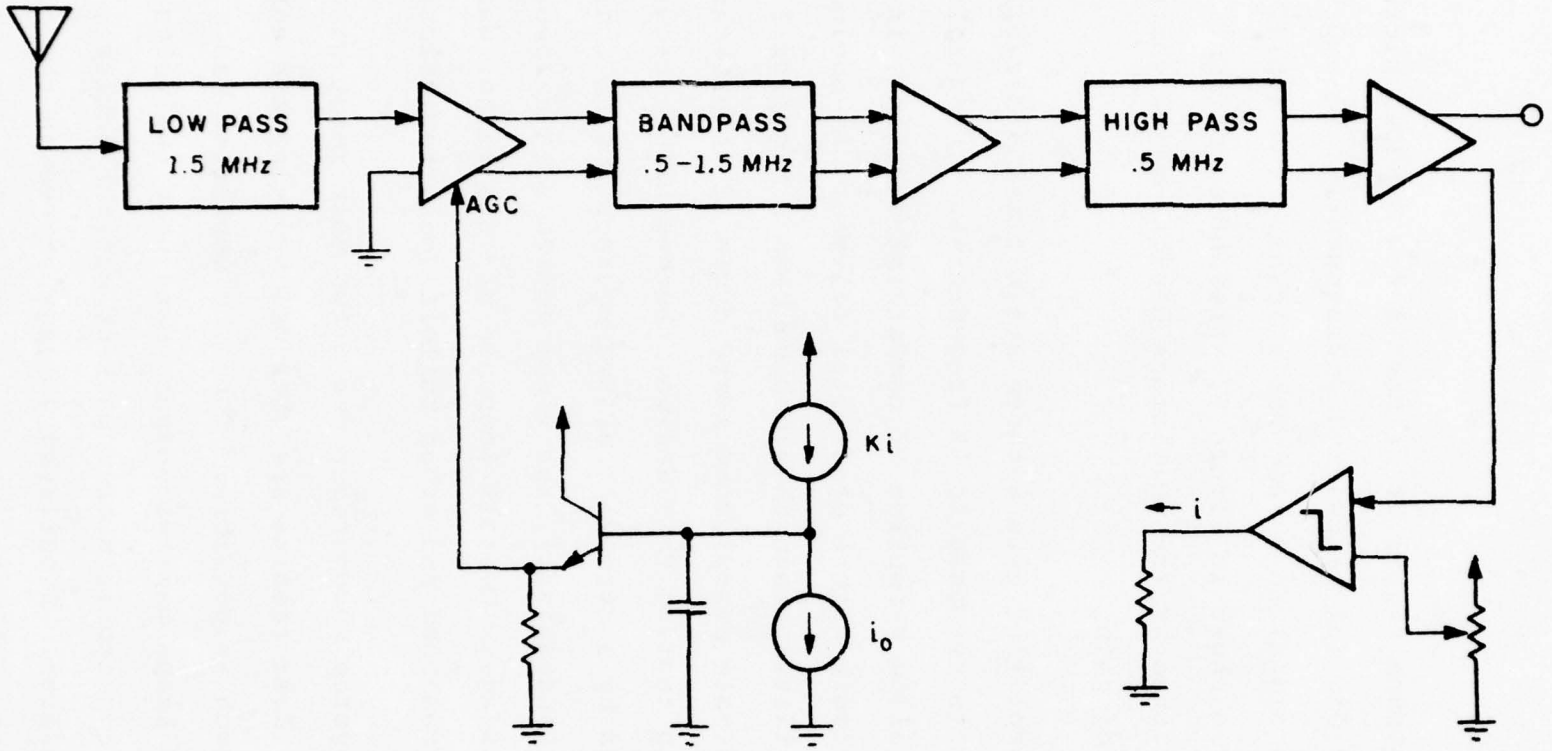


Figure 6. RF Amplifier

Voltage on the capacitor is a function of a constant discharging current source and a comparator controlled charging current source. The comparator output is dependent upon the amplifier output signal and an external two volt DC level.

The charging current exceeds the discharging current by a small amount. Thus, as long as the amplifier output remains at a level below two volts the gain will increase. If the output exceeds two volts, the charging current is shut off and the gain rapidly decreases. By adjusting the DC level and the ratio of the current sources, an output can be obtained that is near optimum for the encoder.

The time constant of the feedback loop can be changed by adjusting the magnitude of the current sources and the size of the capacitor. They are set for a rise time of twenty seconds. and a fall time of five seconds. This allows for correction of signal strength variation due to antenna movement and weather conditions. But changes in the signal due to modulation are ignored.

Signal Sampling

In order to use a lowpass filter to extract the desired station from the AM band, the input signal is sampled at the carrier frequency. Figure 7a shows how the AM band might look with three stations present. In order to receive the 1MHz station, the sampling rate is chosen as 1MHz. The spectrum after sampling is shown in Figure 7b. The original spectrum, folded around the 500KHz folding frequency, is repeated at 1MHz intervals. The spectrum is, in reality, much smaller in amplitude after sampling, as the signal is spread out over the frequency spectrum. No attempt was made to scale amplitudes in these figures. Notice that the desired station has been frequency shifted down to DC where it can be lowpass filtered out of the rest of the spectrum.

Figure 7c shows how the sampled spectrum would look for receiving the 1.5MHz station. Figure 7d shows the sampled spectrum for receiving the 750KHz station. Here an obvious problem exists because the sampling frequency is insufficient to prevent folding of the higher frequencies onto the desired station. The result is that second harmonics of low frequency stations will be received as well as the desired station. This problem is corrected in the sampling circuitry.

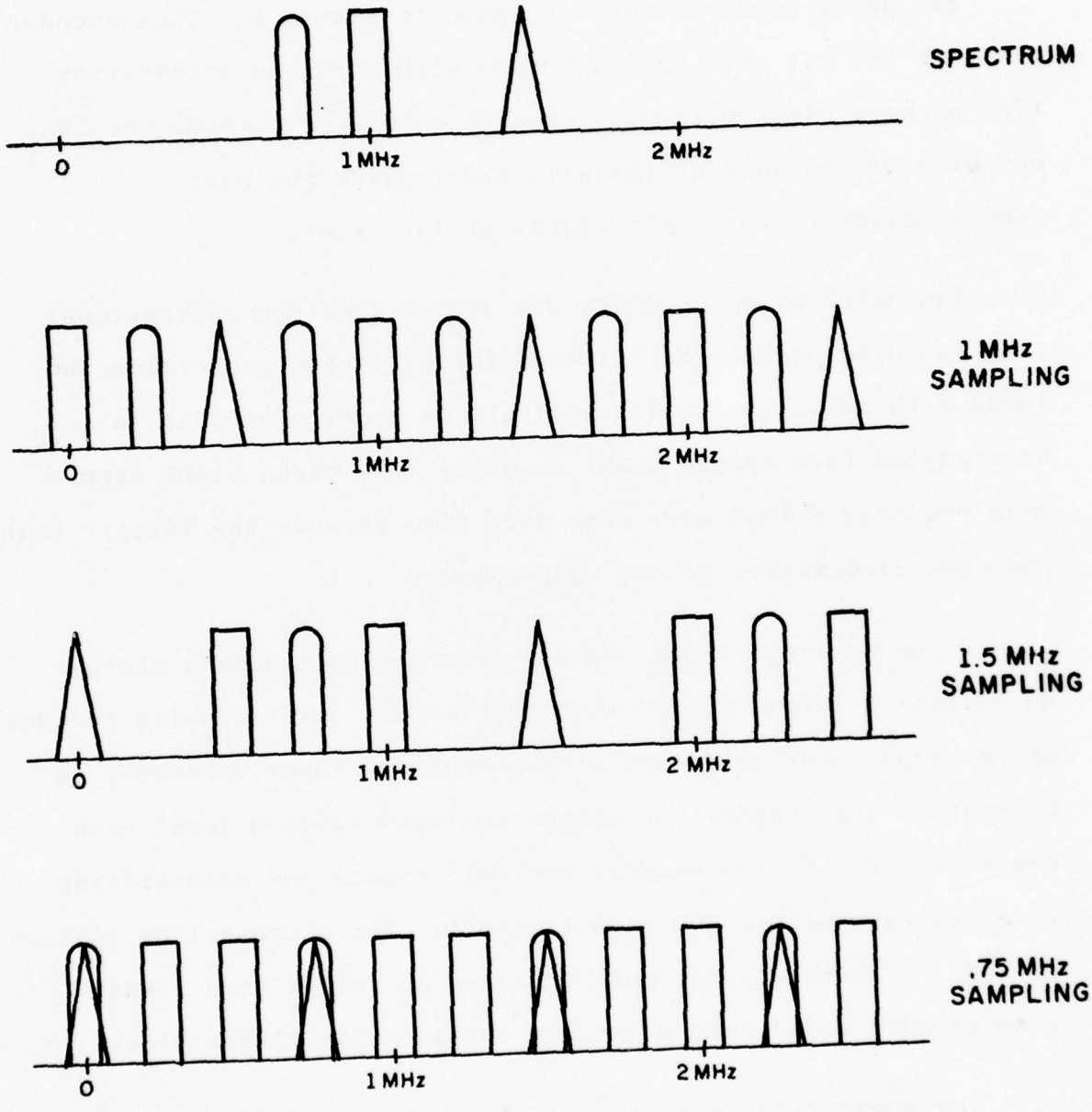


Figure 7. Spectral Effects Of Sampling

Encoding

The delta block encoder is shown in Figure 8. This encoder produces ten bit uncompacted bursts with a finite integration time of nine clock periods. As with compacted bursts, the last ten bits of the data stream will always give the best approximation to the input signal at that time.

The delta block encoding was chosen over the conventional ramp encoding because of its superior filtering properties and lower data rate. Data bits coming from a delta encoder in uncompacted form appear quasi randomly in a given block sample. This property allows more even data flow through the filters than the compacted format of the ramp encoder.

Slope overload in an encoder occurs when the data changes quantization levels faster than the encoder. For a delta encoder the minimum clock speed for a given maximum input frequency is $f_{ck}=\pi Nf$ as the encoder can change one quantization level each clock period. A ramp encoder can only change one quantization level every N clocks and thus $f_{ck}=\pi N^2 f$. For a typical AM station of 1MHz, a delta encoder must function at better than 33MHz; ramp encoder would need a totally unreasonable 330MHz clock.

The above results were derived on the assumption that the maximum slope occurring in a bandlimited signal is the maximum slope of the highest frequency present. In reality this is not the case; the theoretical maximum slope of a wide bandlimited signal approaches infinity when its continuum of frequencies add

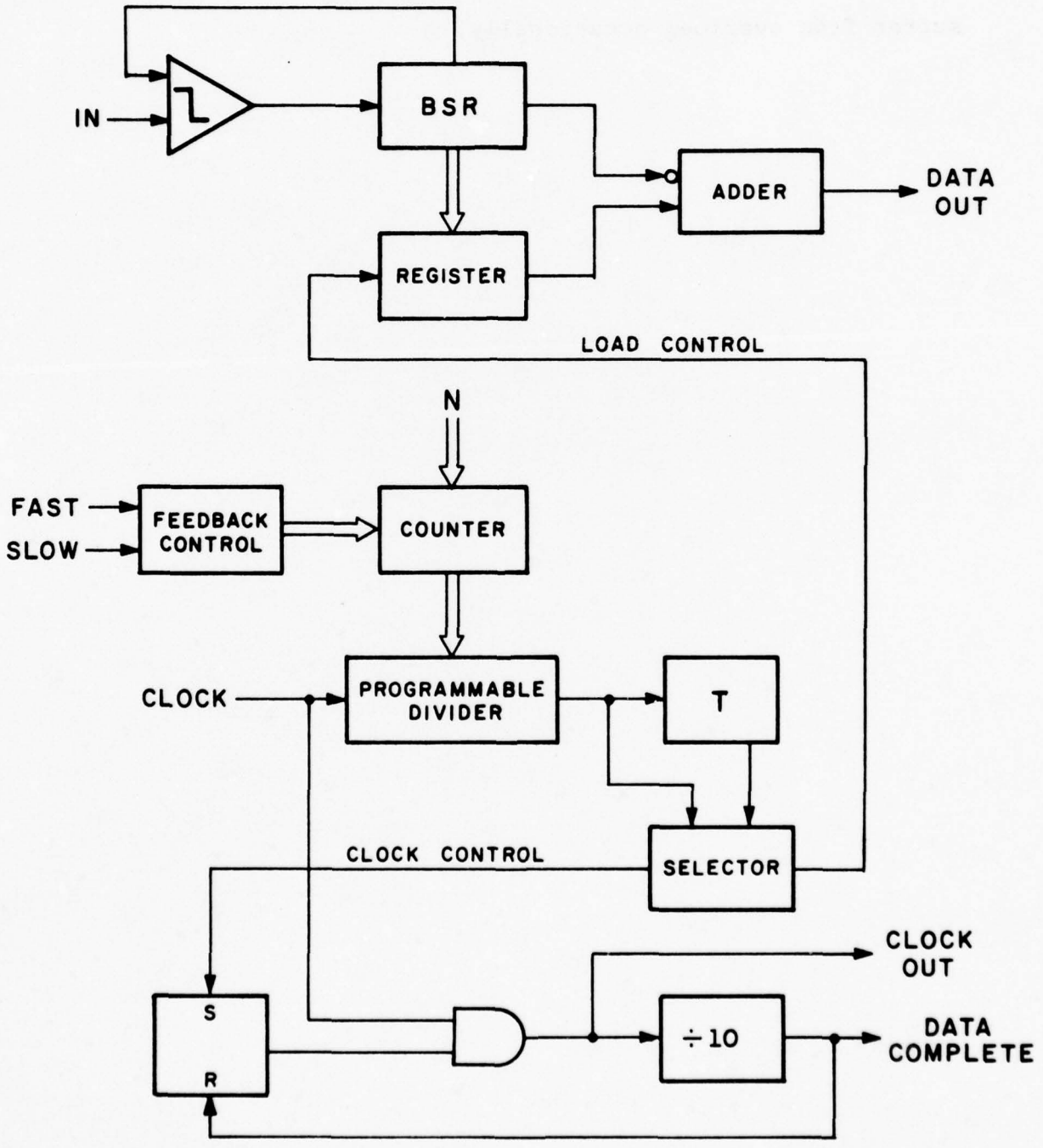


Figure 8. Encoder And Station Selector

signal approaches infinity when its continuum of frequencies add constructively. Obviously this extreme case of slope overload will occur only infrequently, however any practical encoder will suffer from overload occasionally.

Station Selection

A station is selected by sampling at its carrier frequency. The station selector circuitry is shown in Figure 8 along with the encoder. A programmable divider, operating from a 42MHz crystal clock, provides the control signals. Every N clocks the selector alternately enables the load control and clock control lines. During load control a sample is stored in the register. Following a clock control, this sample and the present sample are subtracted, with the result shifted out in a ten bit burst. The ten clock period burst is generated by an AND gate and a divide by ten counter. Once enabled, ten clocks are passed through the AND gate overflowing the counter and resetting the circuit.

Timing diagrams of these signals are shown in Figure 9. Note that the desired station is sampled alternately at positive and negative peaks, while the second harmonic is always sampled at the same position on the waveform. When subtracted, the samples are additive for the desired station, but cancel out the second harmonic.

Depending upon the value of N, various stations can be tuned in. Unfortunately this method of tuning is not readily suitable for commercial AM. The period is varied in a linear fashon with N, resulting in nonlinear frequency steps as opposed to the uniform 10KHz steps of commercial stations. However, several commercial stations can be received by a system using a 42MHz clock. Table 3 lists those stations and the required N, assuming

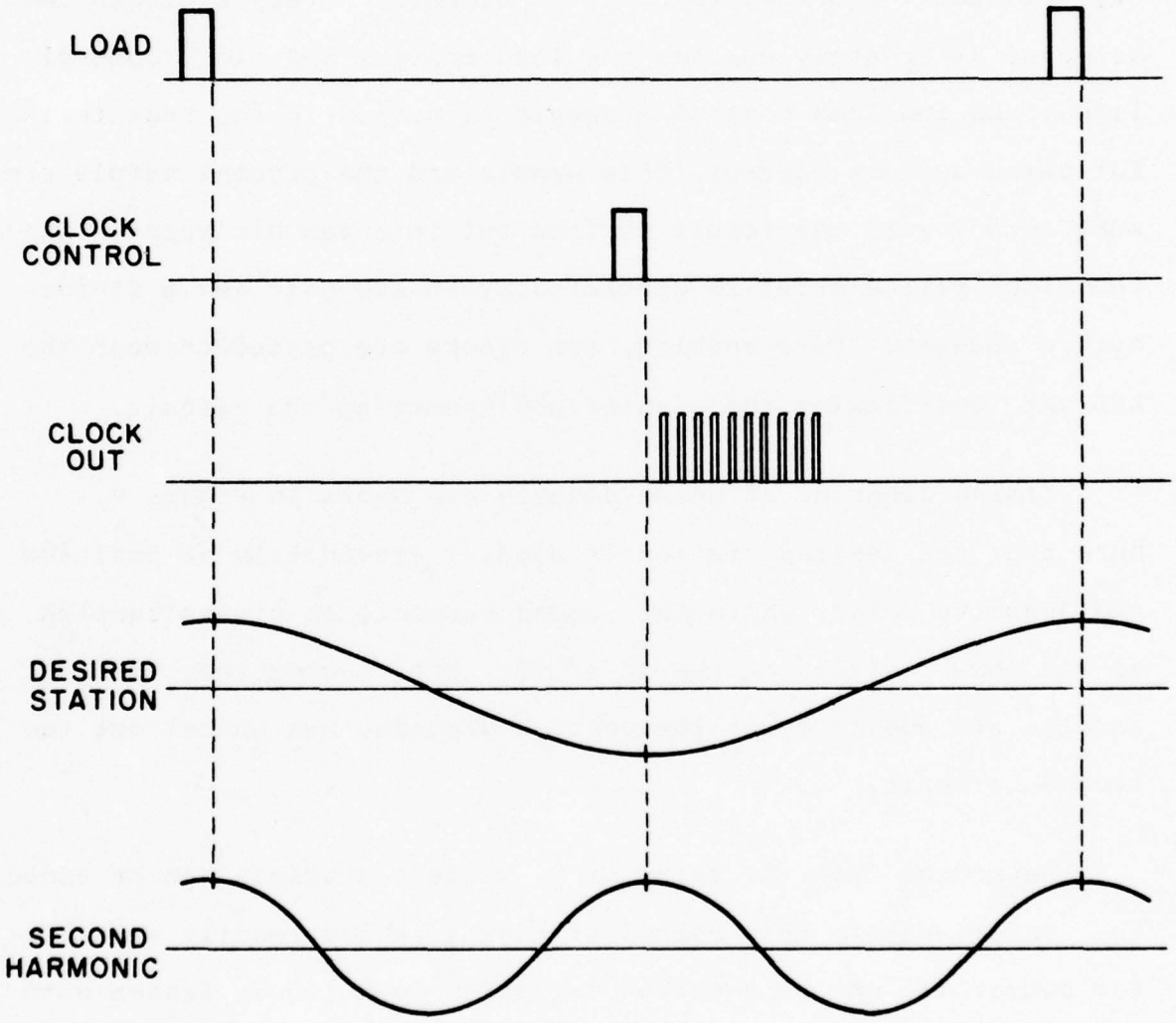


Figure 9. Selector Timing

that the subtractor had been disabled (a feature included in the design).

Table 3. Tunable Am Stations

N	f(KHz)
28	1500
30	1400
35	1200
40	1050
42	1000
50	840
56	750
60	700
70	600
75	560

Inputs from the feedback circuitry enter the selector as either a fast or a slow signal. They cause the counter to count up to N+1 or down to N-1 for one cycle only of the programmable divider. The counter then reloads itself with the external input N. The effect is a movement of the sampling point on the carrier by 1/N in the appropriate direction from the peak. These correction signals are received from the feedback circuit at a rate sufficient to correct for station or system clock drift of ten parts per million.

ECL TTL Conversion

The high speeds of the circuits discussed so far necessitated that ECL be chosen in their design. Since TTL is less expensive and easier to work with, its use is desirable in the rest of the radio. A beat down circuit constructed to interface the ECL and TTL is shown in Figure 10a. This block of logic, cutting the data and clock rates by a factor of three, allows both 14MHz TTL and 42MHz ECL operation.

A block of data is shifted into a Schottky TTL shift register at the ECL clock rate. Upon completion of this operation a "data complete" signal is received causing a second, slower register to be parallel loaded. This register is subsequently dumped serially by a ten bit clock burst occurring at the TTL clock rate. This ten clock burst is generated as before only using TTL instead of ECL components.

Division by three in the beatdown circuit was not an arbitrary choice. The TTL clock frequency had to be sufficient to allow processing a ten bit sample between sample times. In other words the TTL clock must run at least ten times the frequency of the highest desired station frequecy. However, this clock can not exceed the radio's upper operating speed of 18MHz. The choice of 42MHz as the system clock rate was based on its ability to match several commercial frequencies as discussed previously. Frequencies much below this would cause excessive slope overload in the encoder.

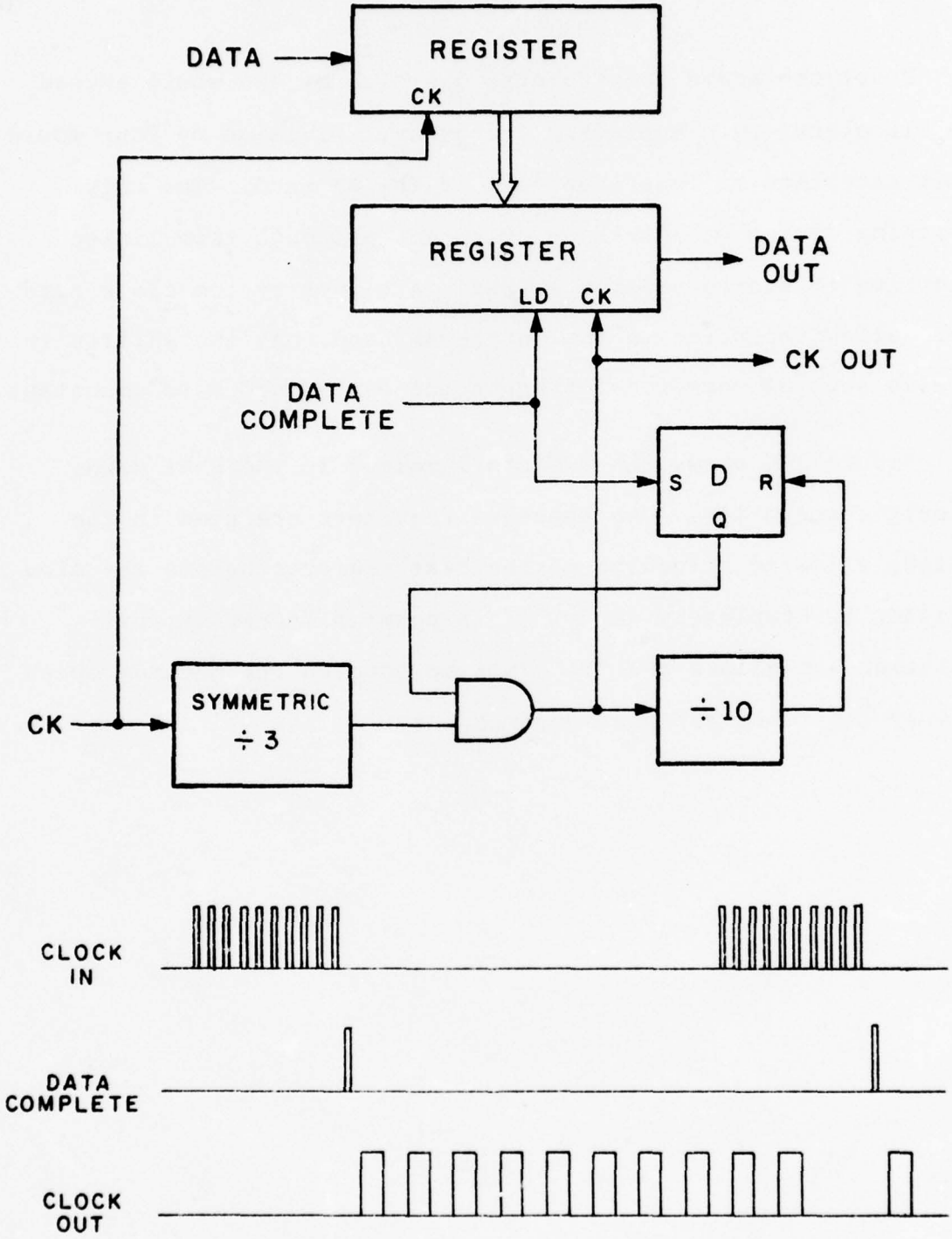


Figure 10. Beat Down Network

Under the above constraints, division by two would exceed the TTL circuitry's operating frequency. Division by four would limit reception to less than half of the AM band. The only remaining choice was division by three, although this limited reception to stations below 1.4MHz. A higher system clock rate would allow reception in the entire AM band, but the ability to receive several commercial frequencies was judged more important.

Figure 10b shows the timings involved in the beat down circuit's operation. Two seperate registers are used in the design, allowing reloading of the fast register before the slow register is completely empty. This permits faster circuit operation and allows the TTL clock to aproach its optimum speed of only ten times the station frequency.

Lowpass Filtering

The most important part of this project is its multistage burst lowpass filter. A recursive filter would be ideal for this application except that, as discussed previously, it is inherently unstable and requires a precision unattainable in Burst Processing. The filter chosen was a fixed impulse response design, although it appears much hardware would be required to achieve the desired selectivity. To illustrate the problem of a FIR filter with high selectivity, consider receiving a station at one megahertz. After sampling, a lowpass filter with about four kilohertz passband is needed. Using a good Hamming weighted filter N stages are required where

$$N = 2/3\ f_{sampling} / f_{cutoff} = \frac{2/3 \times 10^6}{4 \times 10^3} = 167 \quad (18)$$

Thus this design would require 167 ten bit shift registers accompanied by a huge number of adders and multipliers.

Fortunately cascading of stages permits realization of enormous savings in hardware while retaining the filter's excellent selectivity and rejection characteristics. Using four filters, two of which reduce the sampling rate to succeeding stages, results in a rather simple design. The most complex filter contains only fifteen shift registers and four one bit adders.

Figure 11 shows the response curves of the four filters with the radio tuned to 1MHz. Filter A, a simple high speed divide by two network, is constructed from two flip-flops and a few gates. By dividing both the data and clock rates by a factor of two it effectively reduces the sampling frequency seen by the next stage. The sinx/x frequency response of this first filter is displayed in Figure 11a.

Filter B now operating at an effective 500KHz sampling rate is the four stage rectangular weighted design that was shown in Figure 1. Filter C is identical to Filter B in response, but in hardware it is a simple divide down network similar to Filter A. In this case both the data and clock rates are reduced by a factor of four, further lowering the effective sampling rate.

Referring to Figure 11c, the combined response of Filter's A, B, and C is shown. The response of Filters B and C is periodic in 500KHz, resulting in a peak at 500KHz. However, the response of Filter A is zero at this point forcing the overall response to zero.

The height of the peaks on either side of 500KHz is dependent upon the characteristics of the component filters. Sharper rolloff in Filters B and C would reduce the peaks. In contrast, a steeper slope near 500KHz in Filter A would increase peak amplitude. Filter B was required to increase the rolloff of Filter C reducing the peaks to a reasonable value of -30db.

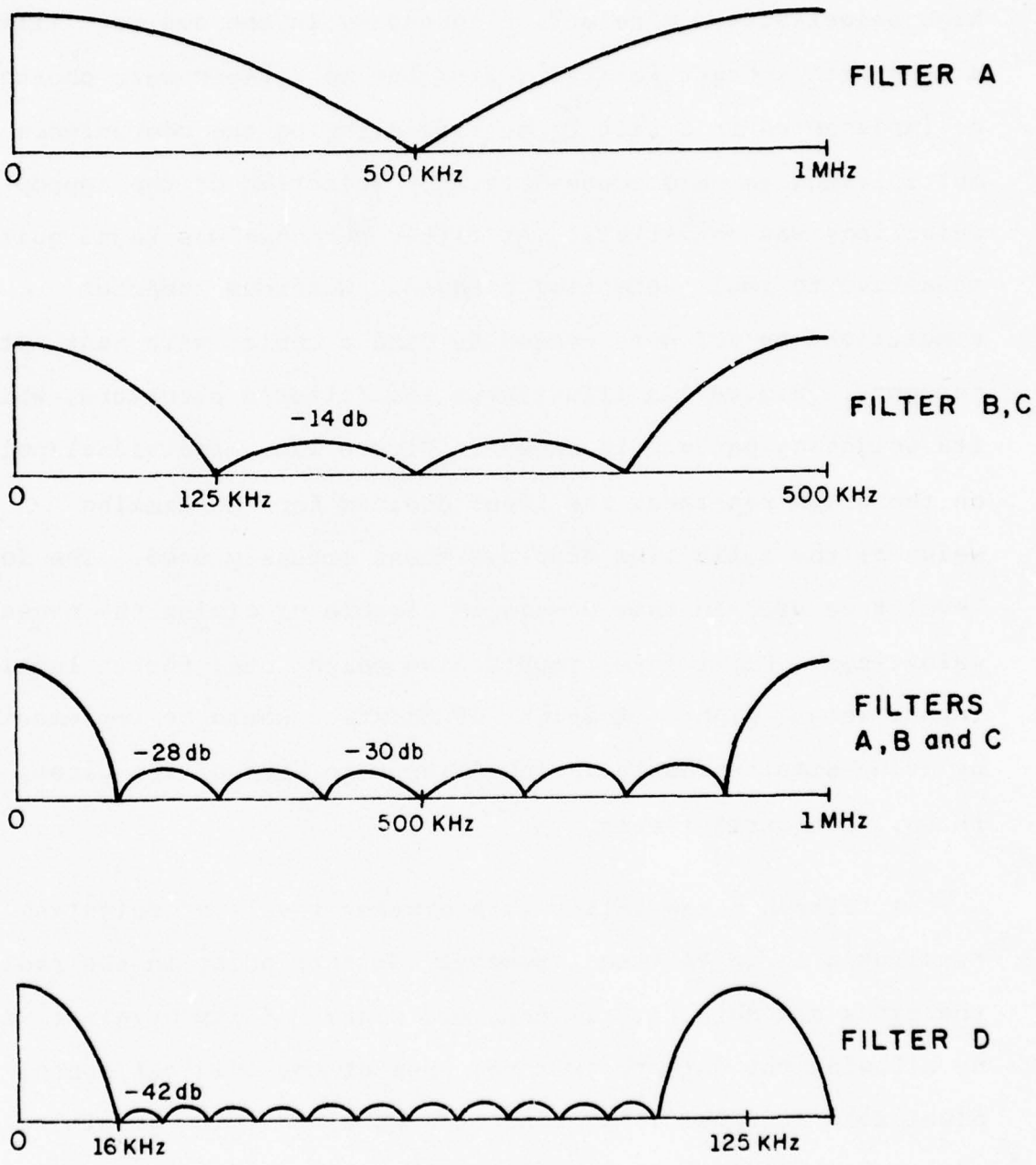
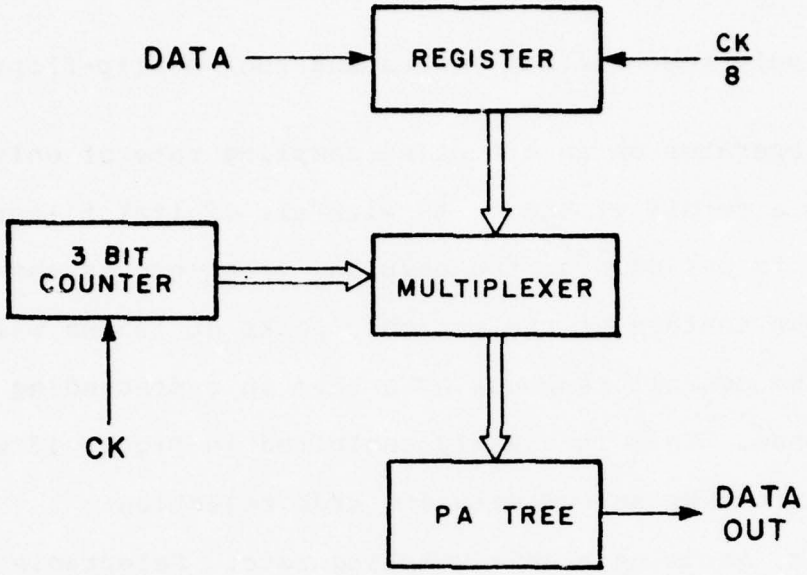


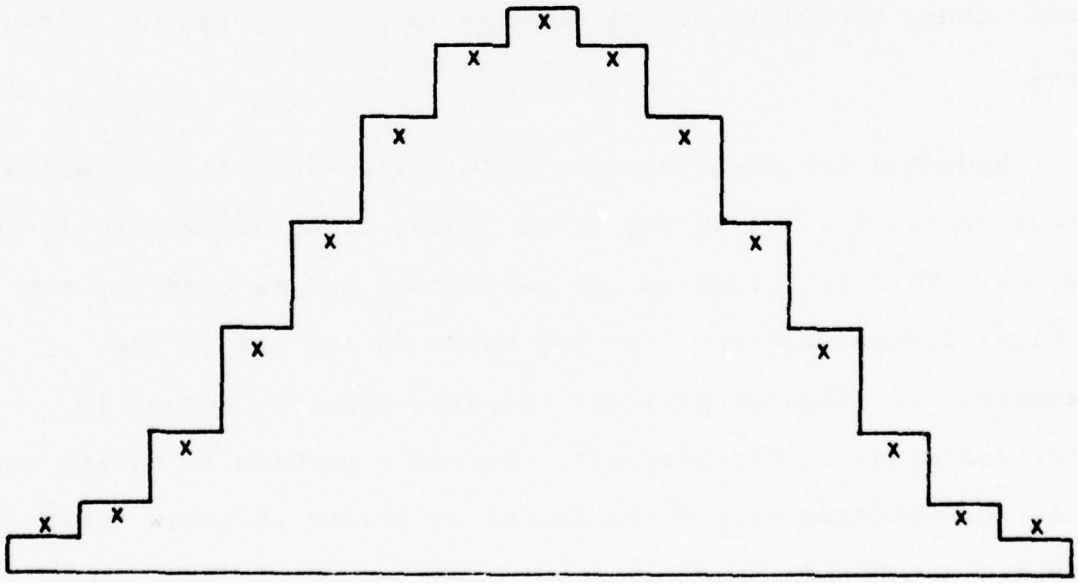
Figure 11. Filter Frequency Responses

The choice of a Hamming weighting for Filter D gives the high selectivity and rejection necessary in the design. Fifteen stages with integer weighting from one to fifteen were chosen to be implemented in direct form, thus allowing the convenience of multiplexing to be discussed later. Selection of the appropriate weightings was not trivial, as filter response was found quite sensitive to small weighting changes. Numerous computer simulations by FFT were needed to find a choice with near optimal response. Figure 12a illustrates the filter's structure, while its weighting pattern is shown in Figure 12b. Individual points on the graph represent the ideal choices for the Hamming weights; the solid line displays those actually used. The four level tree used in this design is capable of giving the necessary weightings. First level imputs give weight one; fourth level inputs weight eight. A weight of thirteen would be implemented by using simultaneously a single input in each of the first, third, and fourth levels.

A fifteen stage filter with sixteen levels of weighting requires a large PA tree. However, at this point in the radio, the clock and data rate is only one eighth of its original value. By allowing the data to exit the tree at the original speed, an eight fold increase in precision is obtained. The resultant eighty bit sample is warranted by the good selectivity and rejection characteristics of the filter. Using this approach also allows the PA tree to be multiplexed with a concurrent reduction in its complexity by a factor of eight. Overall the PA



(a) Implementation



(b) Weighting Function

Figure 12. Hamming Weighted Filter

tree contains only four one bit adders and four D flip-flops.

Filter D operates on an effective sampling rate of only 125KHz and has a cutoff of 4KHz. As with all digital filters its response curve is periodic at the sampling frequency as shown in Figure 11d. The centers of the periodic peaks at 125KHz will be cancelled in the overall response by a zero in a preceeding filter's response. This is clearly exhibited in Figure 13 which shows Logituner's 4KHz selectivity and 40db rejection characteristics, assuming a 1MHz sampling rate. Selectable bandwidths of 8KHz and 4KHZ are obtained by changing Filter A from a divide by two to a divide by four circuit. As actual bandwidth for a given station is dependent on the sampling rate used, these bandwidth selections may vary over a factor of two range.

Reducing the sampling rate with divide down filters allows considerable savings in the total amount of hardware used in the design. This technique is not unique; it can be used for any digital lowpass filter. The key point in its use is the canceling of unwanted periodic response peaks by zeroes in previous stage filter response. The only problem is making sure that the response near these canceling points is below their maximum acceptable level.

A simple four stage triangular filter followed by a selective lowpass filter of ten or more stages can be preceded by a divide down filter of any order. A tradeoff between the order

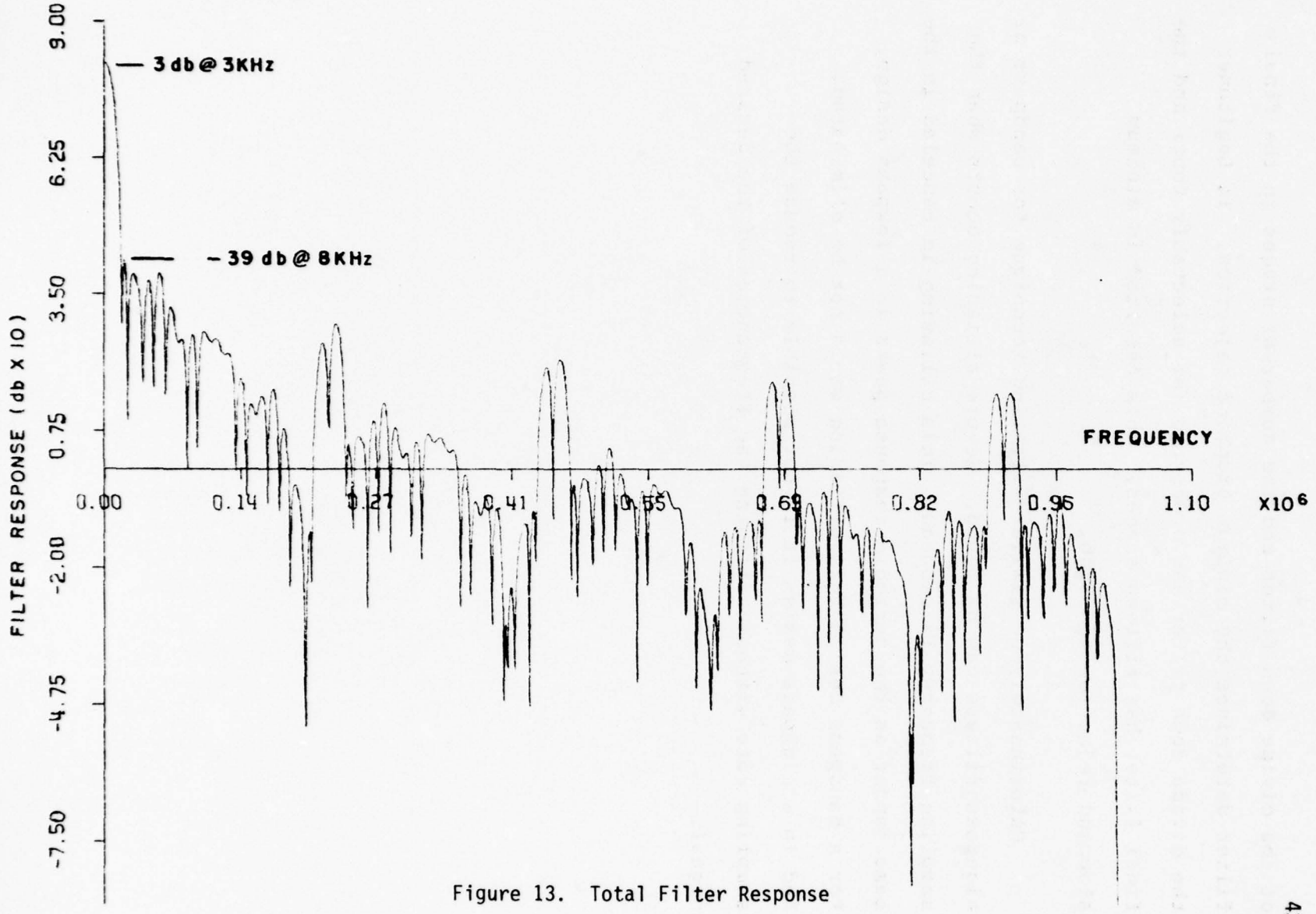


Figure 13. Total Filter Response

of the divide down filter and the number of stages in the final filter determines the minimum stopband rejection. In Logituner the divide down filter has order two (or selectably four) and the final filter has fifteen stages. This resulted in minimum stopband rejection of -56db.

Unfortunately no similar filtering technique for bandpass or highpass filters can be found. Severe ailiasing occurs when the sampling frequency is reduced. This ailiasing is canceled in the same manner as the periodic response peaks in a lowpass design. For a bandpass design this ailiasing would not be eliminated. And in a highpass design it is not possible to reduce the sampling rate without ailiasing the frequencies of the desired signal.

Decoding

The decoder circuit of Figure 14 routes the data stream into a seven bit binary counter. This counter is cleared periodically after storage of its contents in a latch. Four levels of audio gain are digitally achieved by multiplexer shifting the counter contents before their storage in the latch. The contents of the latch control a digital to analog converter, which in turn drives the speaker through an audio amplifier.

The choice of burst to analog conversion by binary D to A was made out of convenience. The simplist burst decoding is realized by a 5KHz lowpass RC filter. The audio information is extremely low in frequency compared to the data rate allowing excellent decoding. However, Logituner's digital filter is a lowpass design. In order to demonstrate that this filter is functioning as promised, no filtering can be present in the decoder.

Final data samples are eighty bits in length. Decoding them using a BSR, the standard burst processing decoder, would require an eighty bit shift register accompanied by eighty current sources. At great savings in circuitry the binary D to A was chosen, although without filtering the D to A output, some noise is introduced into the audio signal.

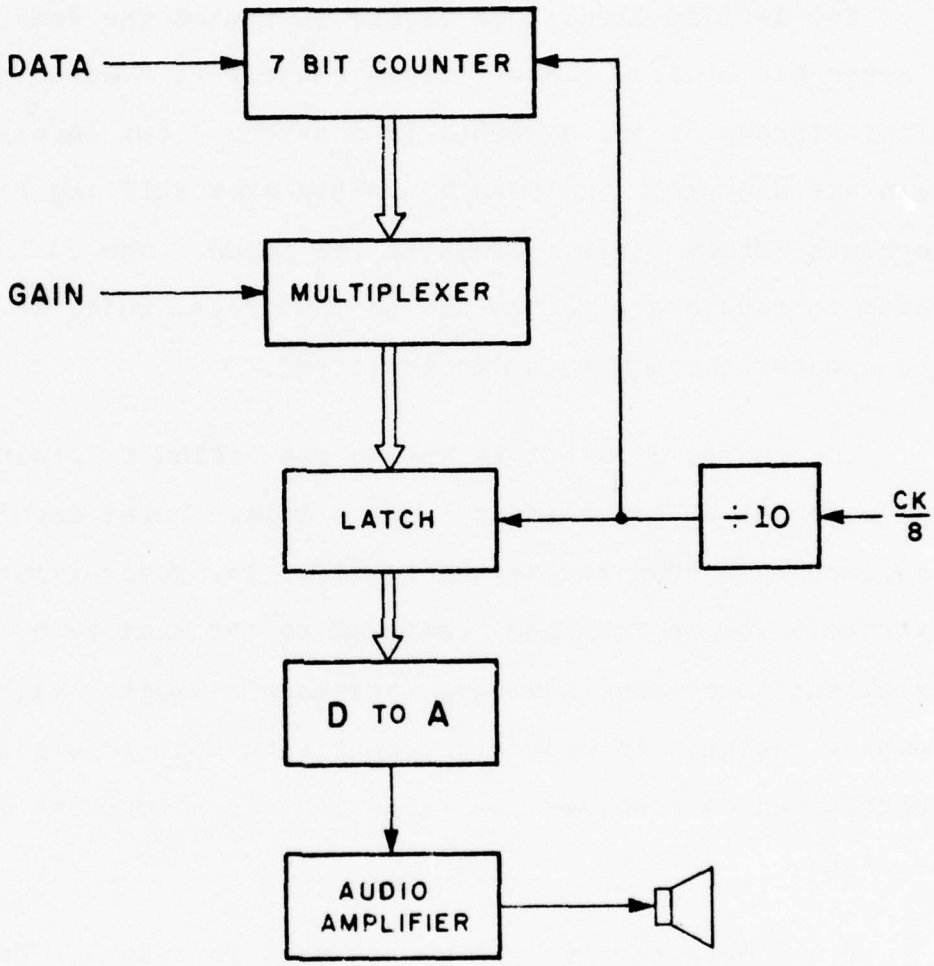


Figure 14. Decoder

Feedback

Logituner's last remaining circuit block is its feedback loop. Sampling must occur without large drift near the carrier peak to obtain a good output signal. The carrier only being identifiable as a DC level at the decoder output makes the feedback loop's task difficult.

A DC filter consisting of a large counter that is dumped at a 25Hz rate extracts the carrier. The sinx/x response of this rectangular weighted filter exhibits a cutoff sufficiently sharp to reject the audio information. Comparing past and present outputs of this filter with a preset DC level determines when and in what direction the sampler is drifting from the optimum sampling point.

In this design the optimum sampling point is set at the preset DC level on the rising edge of the carrier. Four conditions of past and present filter outputs (Y and X) are shown in Figure 15, along with the feedbacks loop's decision in each case. Determining the drift and its direction is achievable by straightforward logic. A sampler that is getting closer to the optimum point requires no correction (Figures 15b and 15c). If the sampler is moving away from the optimum value in the decreasing direction, the sampler is running too slow (Figure 15a). If the sampler is running too fast (Figure 15d) it will be moving away from the optimum value in the increasing direction.

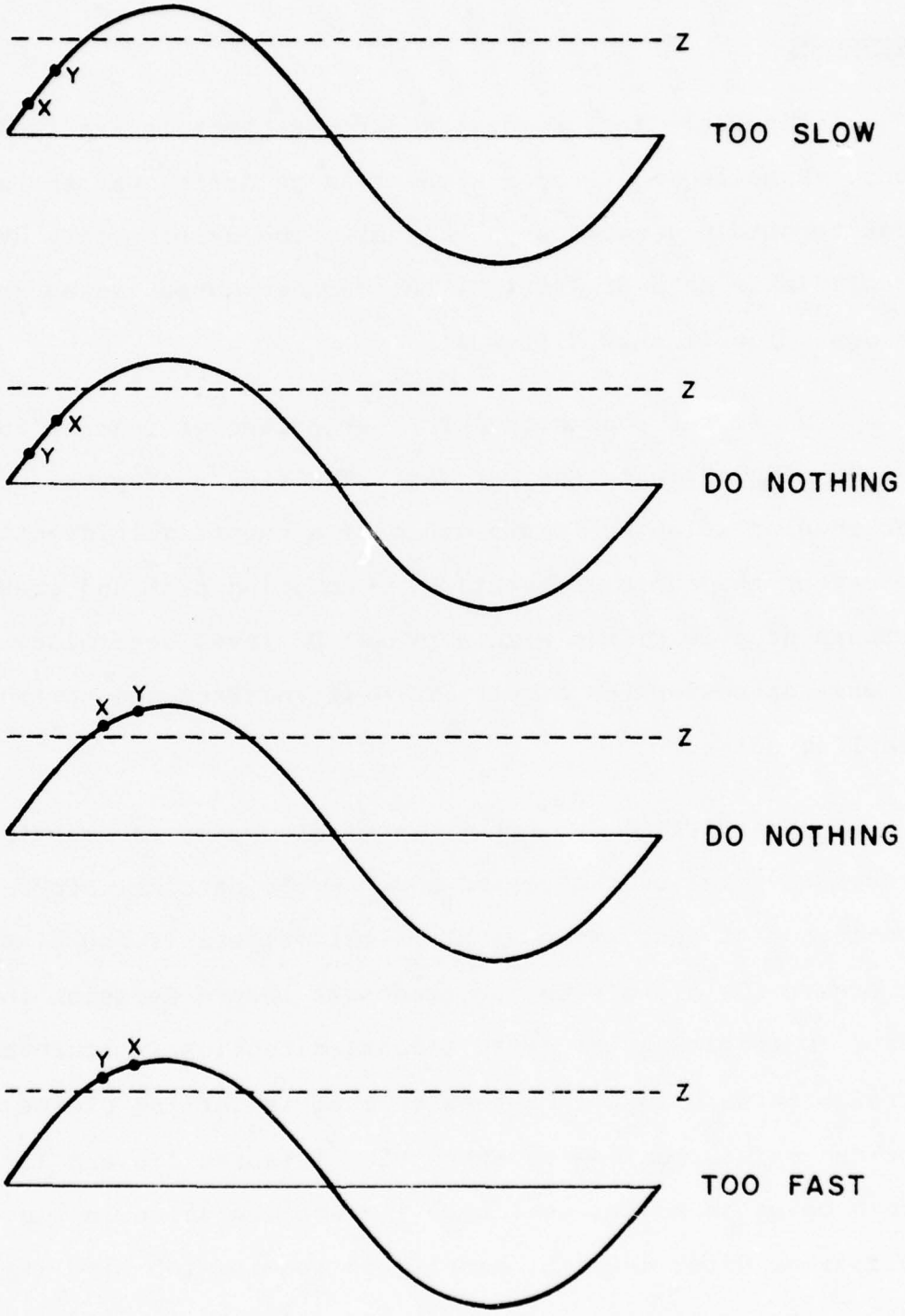


Figure 15. Possible Sampling Conditions

The choice of DC level given to the system depends on the strength of the received carrier signal. A small setting will always keep the station in lock whereas a large setting may exceed a weak carrier's level, losing the station. However, better reception and higher volume will result if the DC level is increased when receiving strong stations.

The feedback circuits are shown in Figure 16. Two latches store past and present values of the sampler's location. Comparators provide the decision box the necessary information. An up-down counter keeps track of the necessary correction, and a rate multiplier provides the necessary correction pulses.

Random drift in the system clock or in the carrier frequency may require only a single correction. However, if the carrier and system clock drift off by a fixed amount, a steady state correction is necessary. The feedback loop is limited to 25Hz update rate by the DC filter. This is much too slow for handling a steady state correction greater than one part per million. The up-down counter included in the design coupled with a rate multiplier allows the sending of multiple correction pulses between updates. A greater range of drift can thus be mastered. The feedback network can handle frequency differences of up to ten parts per million, which covers the FCC tolerances set for commercial stations.

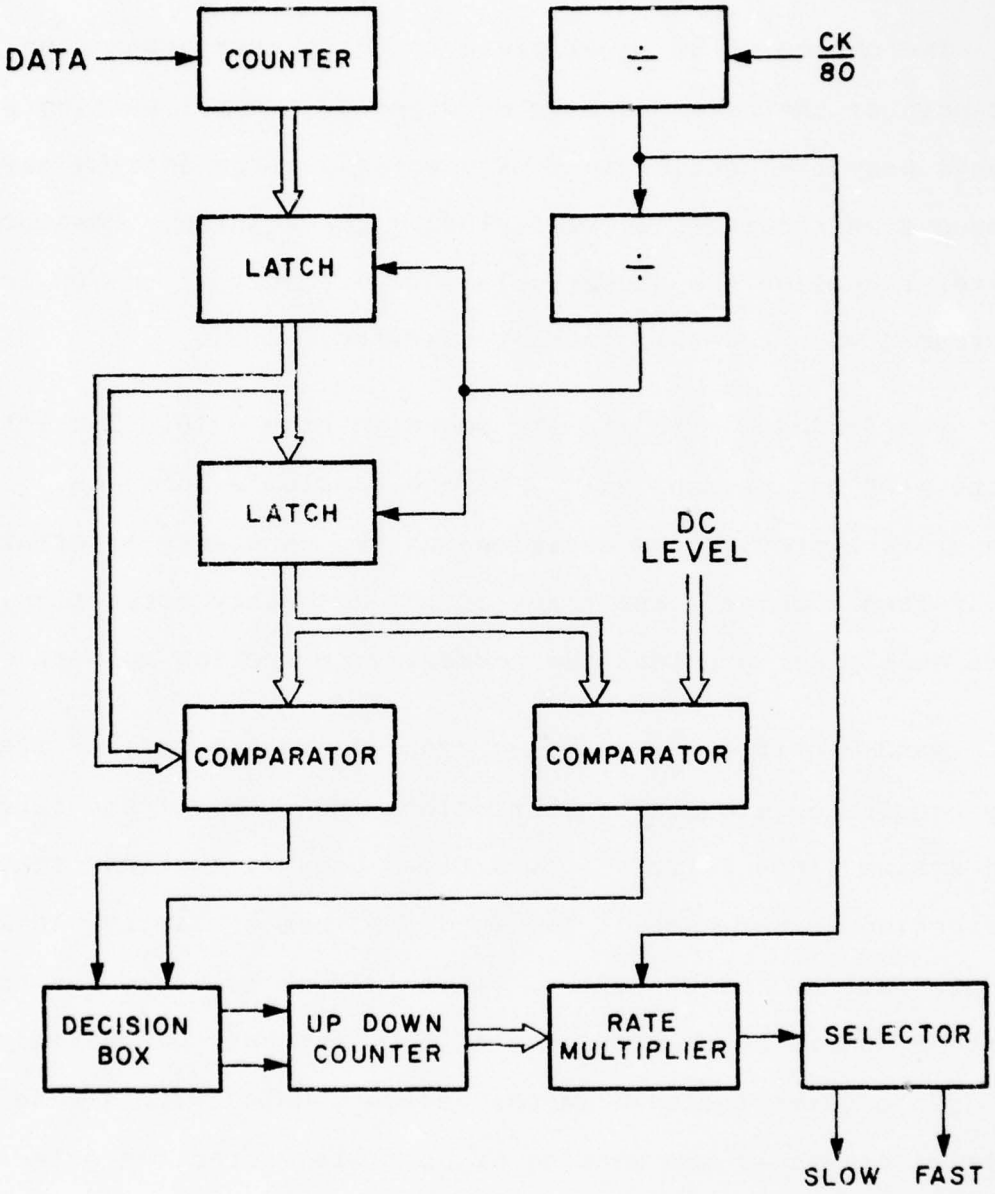


Figure 16. Feedback Circuit

Locking onto a new station requires a significant amount of time due to the feedback's slow update rate. Typically three or four seconds are required to obtain station lock. Once locked the strongly lowpass feedback network will not get lost unless the station drifts out of range.

Noise Analysis

Logituner's lowpass filter response was found by Equation 6 to be

$$h^2 = 1.05 \times 10^{-2} \tag{19}$$

for a white noise input. As Logituner's selectivity is about one percent, this response is reasonable. By computer simulation the noise introduced by the stored carry effect was found to be:

$$N_S = 2.4 \times 10^{-5} \tag{20}$$

Relative figures are being used for these calculations as actual noise power is a function of system voltage levels. Input quantization noise from Equation 2 is:

$$N_Q = 8.33 \times 10^{-4} \tag{21}$$

This noise is filtered and appears at the output as

$$N_{Q_{out}} = N_Q h^2 = 8.8 \times 10^{-6} \tag{22}$$

Under typical operating conditions, Logituner will see an approximately white noise AM band with the desired station contributing approximately ten percent of the total power

received at the antenna. In this situation

$$S_{in}/N_{in} = -10db \tag{23}$$

The undesired stations (assumed to be white noise) will be filtered by the transfer function of the receiver. Assuming a peak voltage at the antenna of four times its RMS voltage and a ten bit burst:

$$\begin{aligned} S_{in} &= .125 \\ N_{in} &= 1.25 \end{aligned} \tag{24}$$

and

$$\frac{S_{out}}{N_{out}} = \frac{S_{in}}{h^2 N_{in} + N_{Q_{out}} + N_s} = 11db \tag{25}$$

Conclusion

Obviously, as designed Logituner does not exhibit an exceptional signal to noise ratio when operated as a receiver. From quantization and internal noise sources Logituner exhibits a signal to noise ratio of

$$\frac{S_{out}}{N_{Q_{out}} + N_s} = 35db \qquad (26)$$

For stronger stations a signal to noise ratio of 45db is approached. Logituner thus has application in an environment where few stations are present in a relatively wide frequency band.

Alternatively, Logituner could be equipped with a selectable bandpass RF amplifier. This amplifier could be inexpensively constructed with poor specifications by relying on Logituner's digital filters for the necessary selectivity.

Logituner is a digital radio is the true sense of the word. With the exception of two analog amplifiers all circuitry used in the design was digital. Even tuning was accomplished digitally in contrast to marketed radios which, while claiming digital tuning, actually use a digitally controlled frequency synthesizer. Although in its present form Logituner will not sweep the market as an AM receiver, it does show some interesting uses of Burst Processing and its application in filters.

8. CONCLUSION

Digital filters have been shown feasible in Burst Processing. They were found to be relatively simple in design and construction. Their noise properties can be analyzed with good results, and a potential application has been demonstrated for their use. The future of burst filters is, at present, certainly bright.

9. LIST OF REFERENCES

1. Poppelbaum W. J. "Statistical Processors," "Advances in Computers," Vol. 14, pp. 187-230, 1976.

2. Bracha, Ehud "Burst Calc (A Burst Calculator)," Report UIUCDCS-R-75-769, Department of Computer Science, University of Illinois, Urbana, Il. October 1975.

3. Mohan, Bracha, and Liu, "Performance Evaluation of the Digital AM Receiver," Report UIUCDCS R-75-757, Department of Computer Science, University of Illinois, Urbana, Il. April 1975

4. Mohan, "The Application of Burst Processing to Digital FM Reveivers," Report UIUCDSC-R-76-780, Department of Computer Science, University of Illinois, Janurary 1976.

5. Faiman , M. Editor, "Annual Progress Report", Department of Computer Science, University of Illinois, Urbana, Il. 1975-76, pp. 1-98.

6. Oppenheim, A. V., and R. W. Schafer, Digital Signal Processing, Prentice-Hall, Englewood, New Jersey, 1975.

7. Rabiner, L. R. and B. Gold, Theory and Application of Digital Signal Processing, Prentice-Hall, Englewood Clifs, New Jersey, 1976.

8. Poppelbaum W. J. "Application of Stochastic and Burst Processing to Communication and Computing Systems," Department of Computer Science, University of Illinois March 1976.

9. Taylor, Gary L. "An Analysis of Burst Encoding Methods and Transmission Properties," Report UIUCDCS-R-75-770, Department of Computer Science, University of Illinois, Urbana, Il., December 1975.

10. Liu, Bede, "Effect of Finite Word Length on the Accuracy of Digital Filters-A Review," IEEE Transactions on Circuit Theory Vol. CT-18 (November 1971) pp. 670-677.

11. Jackson, L. B. "An Analysis of Roundoff Noise in Digital Filters," Sc. D. Thesis, Stevens Institute of Technology, Hoboken, New Jersey, 1969.

12. Bennett W. R. "Spectra of Quantized Signals," Bell System Technical Journal Vol. 27, 1948 pp.446-472.

APPENDIX
CIRCUIT DIAGRAMS

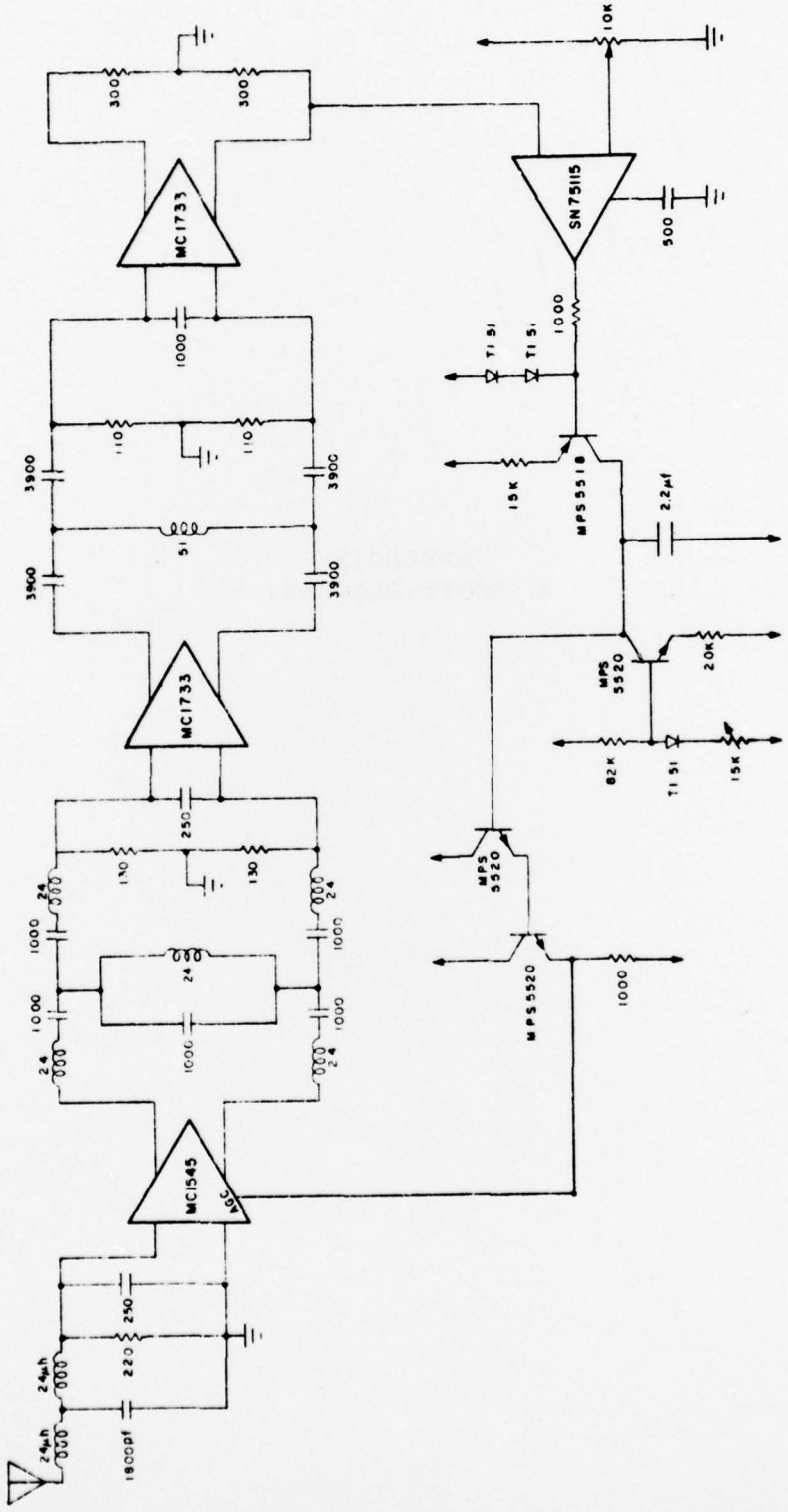


Figure A1. RF Amplifier

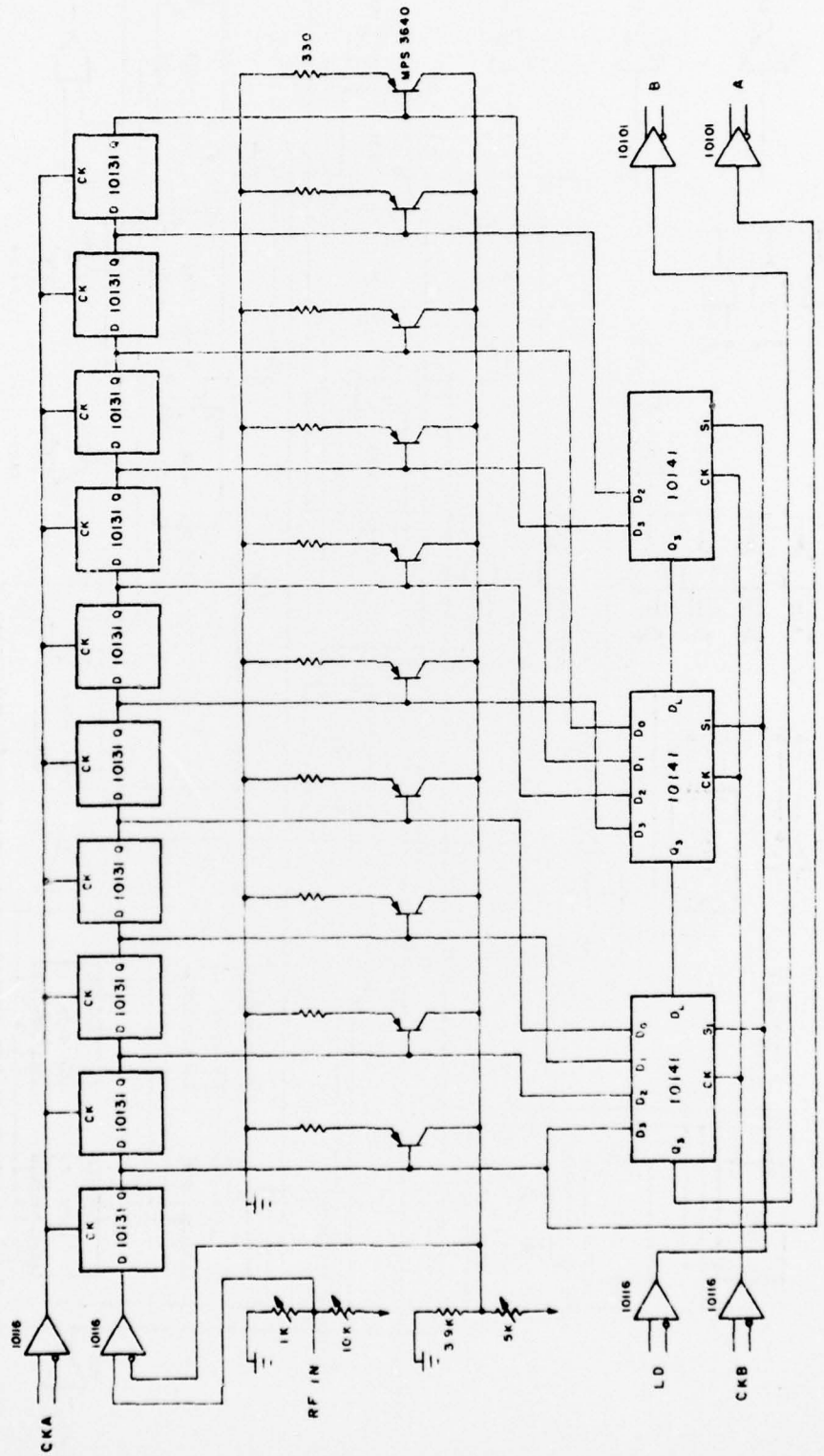


Figure A2. Encoder

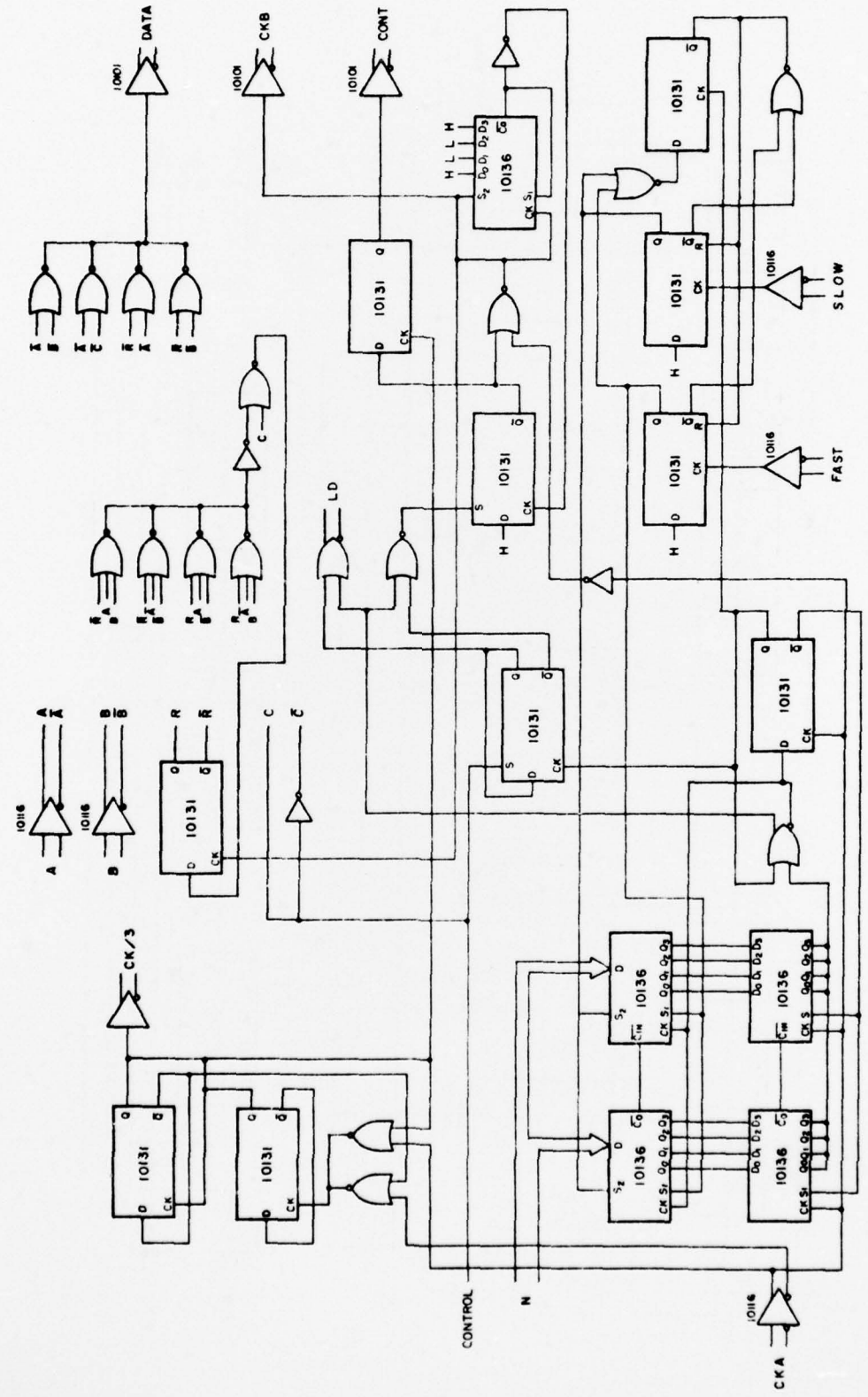


Figure A3. Station Selector

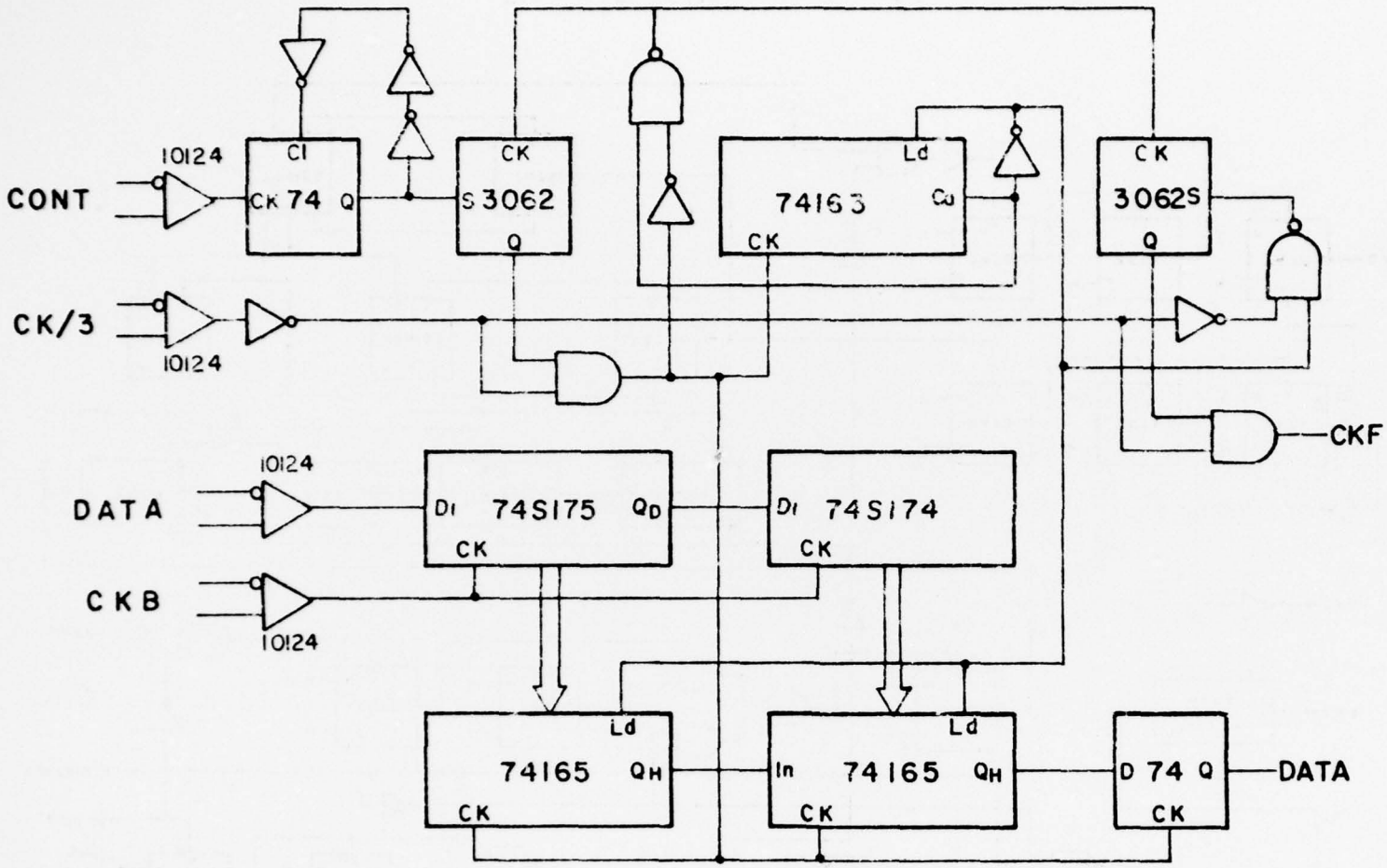


Figure A4. Beat Down Network

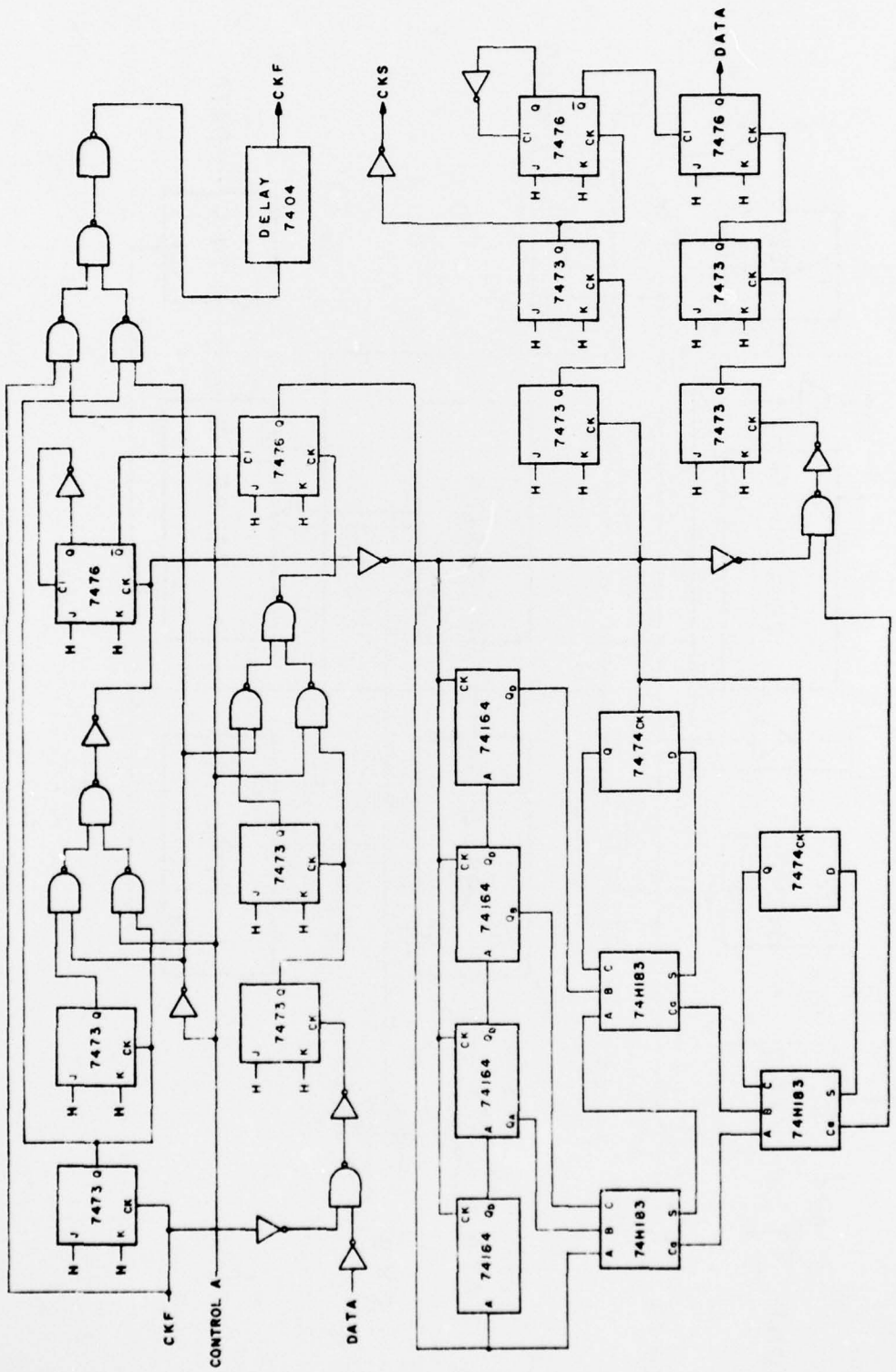


Figure A5. Filters A, B, and C

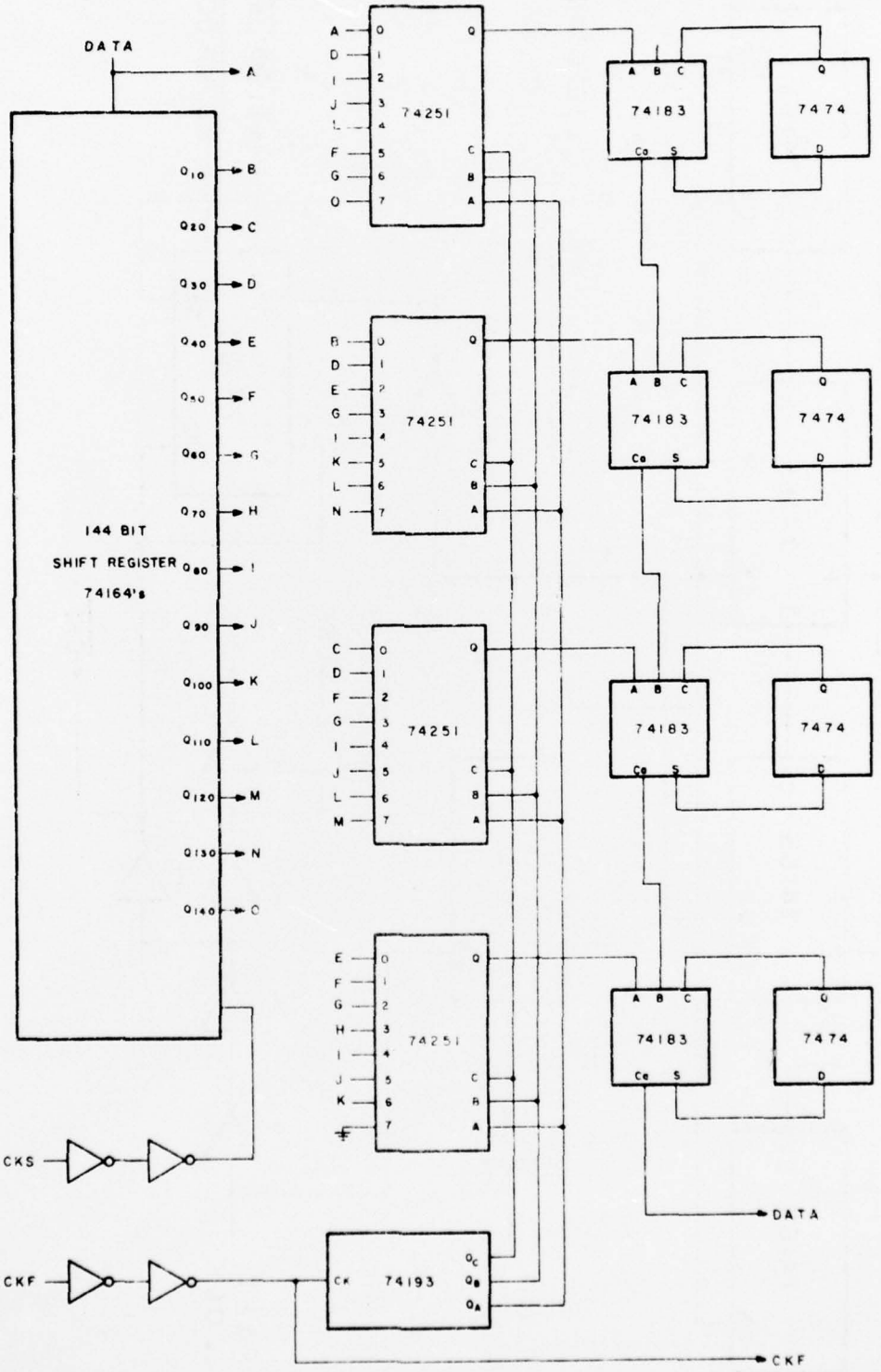


Figure A6. Hamming Weighted Filter

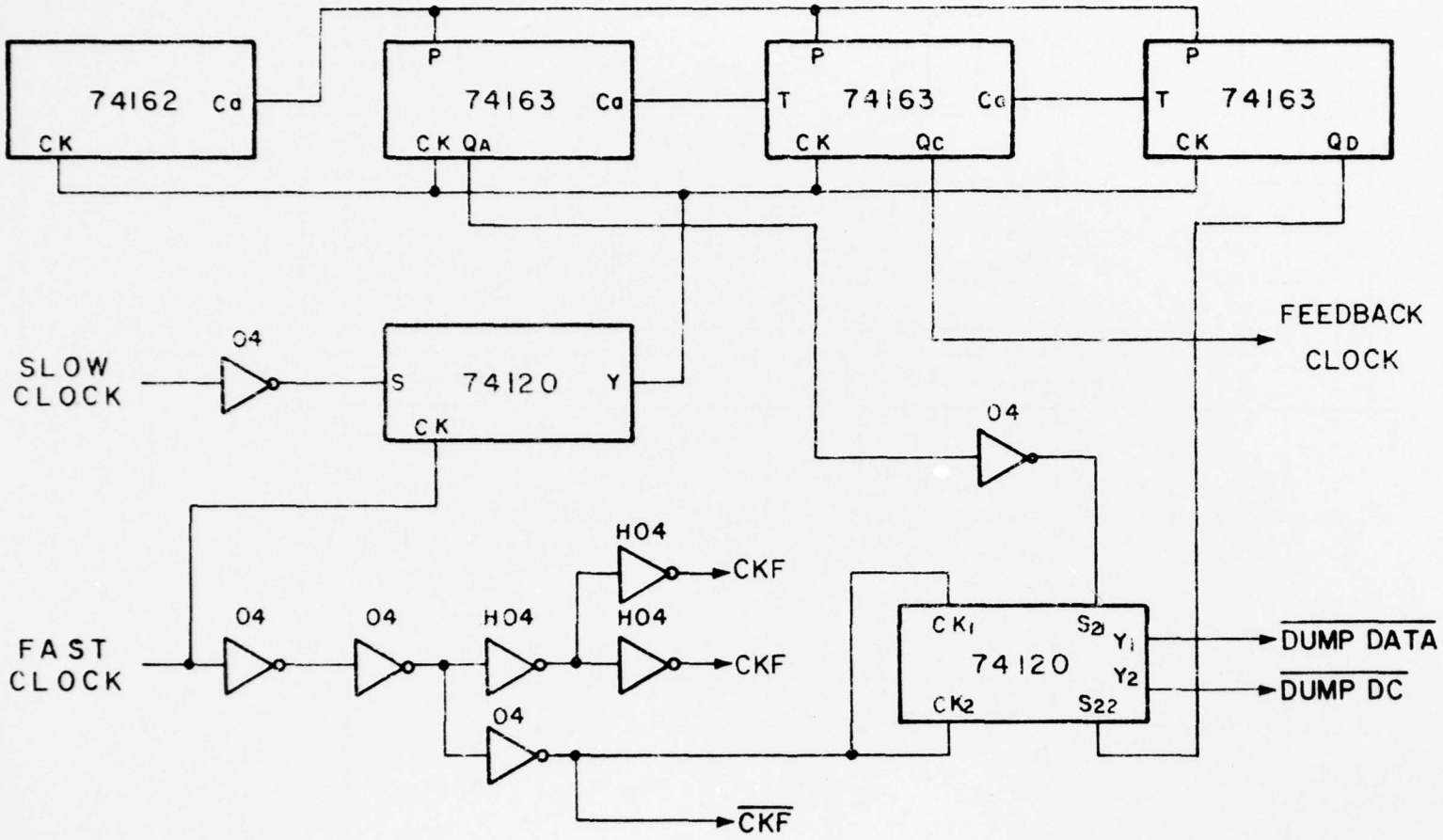


Figure A7. Decoder and Feedback Controls

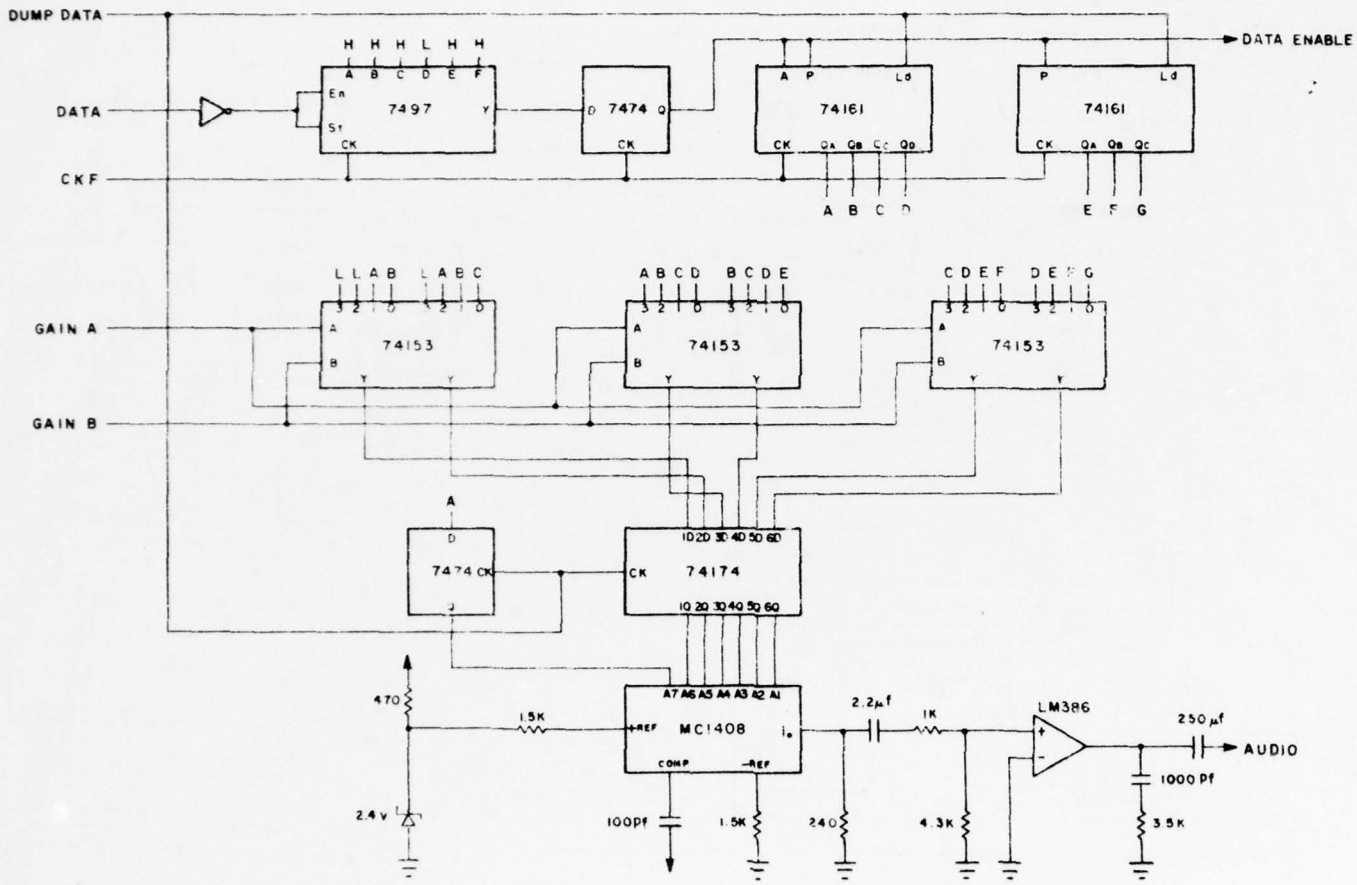


Figure A8. Decoder

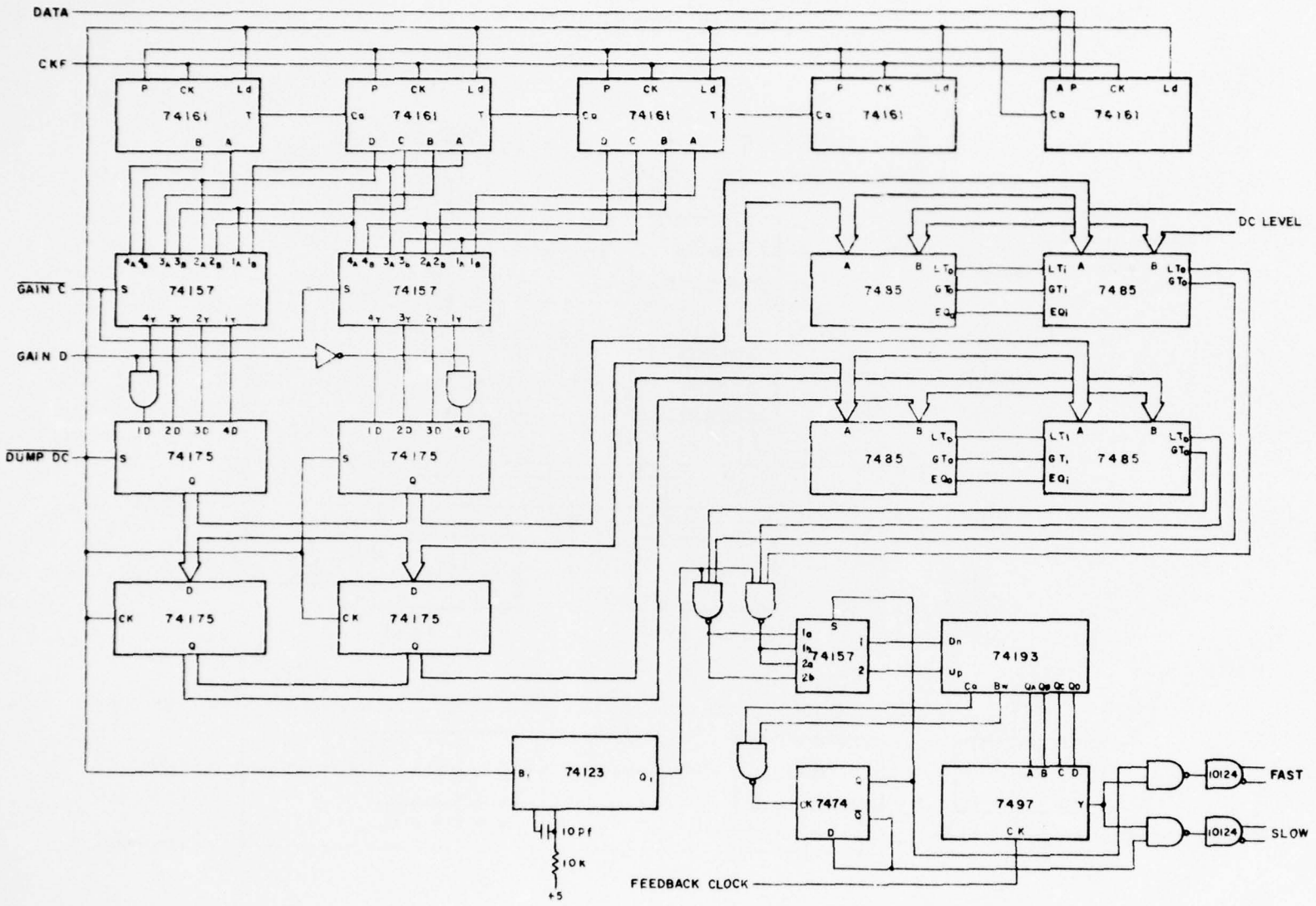


Figure A9. Feedback Circuit

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: UIUCDCS-R-77-871	2. GOVT ACCESSION NO.: UILU-ENG-77-1736
3. RECIPIENT'S CATALOG NUMBER: Master's thesis	
4. TITLE (and Subtitle): DIGITAL FILTERING USING BURST PROCESSING TECHNIQUES.	5. TYPE OF REPORT & PERIOD COVERED: M.S. Thesis/July, 1977
	6. PERFORMING ORG. REPORT NUMBER: UIUCDCS-R-77-871
7. AUTHOR(s): David Kevin Wells	8. CONTRACT OR GRANT NUMBER(s): N00014-75-C-0982
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Office of Naval Research, Code 437, Arlington, Virginia 22217	12. REPORT DATE: July 1977
	13. NUMBER OF PAGES: 70
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Release Unlimited
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Burst Processing, Burst Digital Filter, Logituner, Quantization Noise, Perverted Adder Perverted Adder Tree, Divide Down Filter

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This paper covers the use of Burst Processing in digital filtering. Design and analysis of Burst filters is discussed in detail. A digital radio is shown as demonstration of Burst filtering.

176 011

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

BIBLIOGRAPHIC DATA SHEET	1. Report No. UIUCDCS-R-77-871	2.	3. Recipient's Accession No.
4. Title and Subtitle Digital Filtering Using Burst Processing Techniques			5. Report Date July, 1977 6.
7. Author(s) David Kevin Wells			8. Performing Organization Rept. No.
9. Performing Organization Name and Address Department of Computer Science University of Illinois at Urbana-Champaign Urbana, Illinois 61801			10. Project/Task/Work Unit No. 11. Contract/Grant No. N00014-75-C-0982
12. Sponsoring Organization Name and Address Office of Naval Research Code 437 Arlington, VA 22217			13. Type of Report & Period Covered Master's Thesis 14.

15. Supplementary Notes

16. Abstracts

This paper covers the use of Burst Processing in digital filtering design and analysis of Burst filters is discussed in detail. A digital radio is discussed in detail. A digital radios is shown as a demonstration of Burst filtering.

17. Key Words and Document Analysis. 17a. Descriptors

Burst Processing, Burst Digital Filter, Logituner, Quantization Noise, Perverted Adder tree, Divide Down Filter

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Release Unlimited	UNCLASSIFIED	
	20. Security Class (This Page) UNCLASSIFIED	22. Price